

**ŽILINSKÁ UNIVERZITA V ŽILINE**

**FAKULTA ELEKTROTECHNIKY  
A INFORMAČNÝCH TECHNOLOGIÍ**

**Katedra multimédií a informačno-komunikačných technológií**

28260120241018

**VYTVORENIE GUI PRE POPIS METÓD  
SPRACOVANIA SIGNÁLOV**

**2024**

**Adam Petrovič**



ŽILINSKÁ UNIVERZITA V ŽILINE  
Fakulta elektrotechniky  
a informačných technológií

**Katedra multimédií a informačno-komunikačných  
technológií**

## **VYTVORENIE GUI PRE POPIS METÓD SPRACOVANIA SIGNÁLOV**

BAKALÁRSKA PRÁCA

Študijný odbor:	informatika
Študijný program:	komunikačné a informačné technológie
Vedúci bakalárskej práce:	doc. Ing. Patrik Kamencay, PhD.
Pracovisko vedúceho práce:	UNIZA, FEIT, KMIKT, Univerzitná 1, 010 26 Žilina

**Žilina, 2024**

**Adam Petrovič**



ŽILINSKÁ UNIVERZITA V ŽILINE  
Fakulta elektrotechniky  
a informačných technológií

Katedra multimédií  
a informačno-komunikačných technológií



Akademický rok 2023/2024

## ZADANIE BAKALÁRSKEJ PRÁCE

Meno, priezvisko: **Adam Petrovič**

Študijný odbor: **informatika**

Študijný program: **komunikačné a informačné technológie**

Téma bakalárskej práce: **Vytvorenie GUI pre popis metód spracovania signálov**

Pokyny na vypracovanie bakalárskej práce:

1. Teoreticky spracujte a popíšte problematiku spracovania signálov (opis náhodných a deterministických signálov v čase a spektre, korelačná funkcia a jej využitie, prenos signálov lineárnymi systémami, analógové a diskrétné signály, vzorkovanie a kvantovanie, ...).
2. Teoreticky popíšte a aplikujte matematické postupy na opis signálov a identifikujte najmodernejšie technologické výzvy súvisiace s opisom a spracovaním signálov v rámci súčasných a budúcich komunikačných technológií.
3. Navrhnite a realizujte grafické užívateľské rozhranie pre výučbu predmetu Úvod do signálov a systémov (vizualizácia signálov v časovej a frekvenčnej oblasti, ...) a popíšte postup jeho tvorby.
4. Overte funkčnosť navrhnutého užívateľského rozhrania.

Vedúci bakalárskej práce: Kamencay Patrik, doc. Ing. PhD., Katedra multimédií a informačno-komunikačných technológií, FEIT, Žilinská univerzita v Žiline

Dátum odovzdania bakalárskej práce: 13. 05. 2024

ŽILINSKÁ UNIVERZITA V ŽILINE  
FAKULTA ELEKTROTECHNIKY  
A INFORMAČNÝCH TECHNOLOGIÍ  
KATEDRA MULTIMÉDIÍ A INFORMAČNO-  
KOMUNIKAČNÝCH TECHNOLOGIÍ  
Univerzitná 8215/1, 010 26 Žilina  
-1-

prof. Ing. Róbert Hudec, PhD.  
vedúci katedry

V Žiline dňa 31. 10. 2023

# ANOTAČNÝ ZÁZNAM

**Žilinská univerzita v Žiline**

**Fakulta elektrotechniky a informačných technológií**

**Katedra multimédií a informačno-komunikačných technológií**

<b>Typ práce:</b>	Bakalárska práca
<b>Meno a priezvisko:</b>	Adam Petrovič
<b>Akademický rok:</b>	2023/2024
<b>Názov práce:</b>	Vytvorenie GUI pre popis metód spracovania signálov
<b>Počet strán:</b>	43
<b>Počet obrázkov:</b>	53
<b>Počet tabuliek:</b>	0
<b>Počet grafov:</b>	0
<b>Počet príloh:</b>	1
<b>Počet použ. lit.:</b>	31

## **Anotácia v slovenskom jazyku:**

V rámci tejto práce je riešená problematika vytvorenia GUI pre signály a systémy s cieľom poskytnúť interaktívny kurz ako učebnú pomôcku pre predmet Úvod do signálov a systémov. Okrem popisu samotného GUI sa venuje aj teoretickému opisu signálov a systémov, poskytuje prehľad dostupných kurzov a programovacích jazykov, ktoré umožňujú vizualizovať a spracovať signály.

## **Annotation in foreign language (english):**

This thesis addresses the problem of creating a GUI for signals and systems in order to provide an interactive course as a teaching aid for the Introduction to signals and systems course. In addition to describing the GUI itself, it also discusses the theoretical description of signals and systems, and provides an overview of existing courses and programming languages that allow signals to be visualized and processed.

**Kľúčové slová:** Signály a systémy, GUI, MATLAB

**Vedúci bakalárskej práce:** doc. Ing. Patrik Kamencay, PhD.

**Dátum odovzdania ZP:** 13.5.2024

# POĎAKOVANIE

Rád by som týmto vyjadril veľkú vďaku môjmu vedúcemu práce pánovi doc. Ing. Patrikovi Kamencayovi, PhD., za odbornú pomoc a usmernenie pri písaní a tvorbe mojej bakalárskej práce, ako aj za cenné rady a ochotu venovať mi svoj čas.

# ABSTRAKT

## Abstrakt v slovenskom jazyku

Bakalárska práca sa zaoberá vytvorením GUI pre popis metód spracovania signálov. Cieľom práce je vytvoriť GUI, ktoré bude mať formu interaktívneho kurzu a bude slúžiť ako učebná pomôcka pre predmet Úvod do signálov a systémov. V práci je opísané vytvorené GUI a proces jeho návrhu a tvorby. Práca sa okrem samotného GUI venuje aj teoretickému opisu signálov a systémov, prehľadu existujúcich kurzov a programovacím jazykom, ktoré majú implementované funkcie pre vizualizáciu a prácu so signálmi a systémami. Výsledkom práce je responzívne a interaktívne GUI, ktoré ako učebná pomôcka umožňuje študentom lepšie porozumenie metód spracovania a opis signálov a systémov prostredníctvom interaktívnych cvičení a vizualizácií.

**Kľúčové slová: Signály a systémy, GUI, MATLAB**

## Abstract in foreign language (english)

The bachelor thesis deals with the creation of a GUI for describing signal processing methods. The aim of the thesis is to create a GUI that will be in a form of an interactive course and will serve as a teaching aid for the class Introduction to Signals and Systems. The thesis describes the created GUI and the process of its design and creation. In addition to the GUI, this thesis also discusses the theoretical description of signals and systems, provides an overview of the existing courses and programming languages that have implemented functions for working with signals and systems. The result of this work is a responsive and interactive GUI that, as a learning tool, allows students to better understand the processing methods and the description of signals and systems through interactive exercises and visualizations.

**Keywords: Signals and systems, GUI, MATLAB**

# OBSAH

Anotačný záznam.....	i
PodĎakovanie .....	ii
Abstrakt.....	iii
Obsah .....	iv
Zoznam obrázkov a tabuliek.....	vi
Zoznam skratiek.....	viii
Zoznam symbolov.....	ix
Úvod.....	1
1 Teoretický úvod do signálov a systémov.....	2
1.1 Základné rozdelenie signálov .....	2
1.2 Periodické signály v časovej a frekvenčnej oblasti .....	2
1.3 Neperiodické signály v časovej a frekvenčnej oblasti .....	3
1.4 Vzorkovanie.....	5
1.5 Kvantovanie .....	6
1.6 Diskrétné signály v časovej a frekvenčnej oblasti .....	6
1.7 Konvolúcia a korelácia .....	7
1.8 Stochastické signály.....	7
1.9 Systémy.....	8
2 Dostupné kurzy a aplikácie.....	9
2.1 Kurz od MIT .....	9
2.2 Kurz od ČVUT.....	10
2.3 Kurz od Wolfram .....	11
2.4 Android aplikácie.....	12
3 Programovacie jazyky pre spracovanie a opis signálov .....	14
3.1 Jazyk Wolfram.....	14
3.2 Jazyk Python .....	14
3.3 Jazyk MATLAB .....	15
3.4 Jazyk GNU Octave .....	16
4 Návrh a tvorba GUI .....	17
4.1 Vizuálne členenie a štruktúra GUI.....	17
4.2 Grafický návrh teoretickej časti .....	19
4.3 Návrh teoretických testov .....	20
4.4 Implementovanie kódového editora do GUI .....	22

4.5	Grafický návrh príkladov .....	23
4.6	Tvorba video ukážok a interaktívnych ukážok .....	24
4.7	Export vytvorenej aplikácie .....	26
5	Aplikácia Signály a Systémy .....	27
5.1	Úvodné menu .....	27
5.2	Učebné textové materiály .....	28
5.3	Vytvorené testy .....	29
5.4	Video ukážky signálov a procesov .....	30
5.5	Interaktívne ukážky signálov a systémov .....	30
5.6	Signály a systémy v príkladoch .....	35
5.7	Technické problémy a testovanie .....	39
5.8	Porovnanie vytvoreného GUI s bezplatne dostupnými kurzami .....	41
	Záver .....	43
	Zoznam použitej literatúry .....	
	Čestné vyhlásenie .....	
	Prílohová časť .....	
	Zoznam príloh.....	I
	Príloha A .....	II



# ZOZNAM OBRÁZKOV A TABULIEK

Obr. 1.1: Aproximácia periodického obdĺžnikového signálu: a) časová oblasť, b) frekvenčná oblasť. ....	3
Obr. 1.2: obdĺžnikový neperiodický signál: a) v časovej oblasti, b) vo frekvenčnej oblasti. ....	4
Obr. 1.3: Diracov impulz: a) v časovej oblasti, b) vo frekvenčnej oblasti [1]. ....	4
Obr. 1.4: Heavysideov skok [1]. ....	5
Obr. 1.5: Druhy vzorkovania [1]. ....	5
Obr. 1.6: Proces kvantovania [1]. ....	6
Obr. 1.7: Diskrétny signál: a) v časovej oblasti, b) vo frekvenčnej oblasti. ....	7
Obr. 1.8: Bloková schéma systému. ....	8
Obr. 2.1: Kurz od MIT [10]. ....	10
Obr. 2.2: Kurz od ČVUT – možnosť prezerania jednotlivých častí prezentácie [11]. ...	10
Obr. 2.3: Užívateľské rozhranie kurzu Wolfram [13]. ....	11
Obr. 2.4: Interaktívna ukážka – aproximácia obdĺžnikového signálu [13]. ....	12
Obr. 2.5: Formát rovníc a textu v aplikáciách. ....	13
Obr. 2.6: Reklama v aplikáciách. ....	13
Obr. 3.1: Sínusový signál v jazyku Wolfram. ....	14
Obr. 3.2: Sínusový signál vytvorený knižnicami NumPy a Matplotlib v jazyku Python. ....	15
Obr. 3.3: Sínusový signál v jazyku Octave. ....	16
Obr. 4.1: Možnosť prekliknutia medzi záložkami. ....	17
Obr. 4.2: Komponent GUI pri zväčšení okna aplikácie a) bez využitia Grid Layout b) s využitím Grid Layout. ....	18
Obr. 4.3: Princíp členenia jednotlivých kapitol na podkapitoly. ....	18
Obr. 4.4: HTML komponent s HTML dokumentom. ....	19
Obr. 4.5: MATLAB Live Script. ....	20
Obr. 4.6: Pôvodný návrh testov. ....	21
Obr. 4.7: Finálny návrh testov. ....	22
Obr. 4.9: Výpis do konzoly: a) výstup z kódu b) chyba. ....	23
Obr. 4.10: Návrh vyriešeného príkladu. ....	24
Obr. 4.11: Návrh príkladu na precvičenie. ....	24
Obr. 4.12: Návrh video ukážky. ....	25

Obr. 4.13: Návrh interaktívnej ukážky. ....	26
Obr. 4.14: Nainštalovaná aplikácia v MATLAB APPS. ....	26
Obr. 5.1: Úvodné menu GUI s ukážkami signálov. ....	28
Obr. 5.2: Teoretická textová časť GUI. ....	29
Obr. 5.3: Testy v GUI. ....	29
Obr. 5.5: Video ukážka – proces vzorkovania. ....	30
Obr. 5.5: Interaktívna ukážka sínusového signálu. ....	31
Obr. 5.6 Interaktívna ukážka FR. ....	31
Obr. 5.7: Interaktívna ukážka vlastnosti FT – zmena časového merítka. ....	32
Obr. 5.8: Interaktívna ukážka nestacionárneho stochastického signálu. ....	33
Obr. 5.9: Interaktívna ukážka generovania PP. ....	33
Obr. 5.10: Interaktívna ukážka merania chybovosti kanála. ....	34
Obr. 5.11: Interaktívna ukážka linkových kódov. ....	34
Obr. 5.12: Vyriešený príklad z kapitoly DFT. ....	35
Obr. 5.13: Programovací príklad na komplexné čísla. ....	36
Obr. 5.14: Programovacie príklady na vizualizáciu signálov. ....	36
Obr. 5.15: Príklad na FR s vizualizáciou. ....	37
Obr. 5.16: Príklad na FT s vizualizáciou. ....	37
Obr. 5.17: Príklad na krížovú koreláciu s vizualizáciou. ....	38
Obr. 5.18: Príklad na kapacitu kanála. ....	38
Obr. 5.19: Príklady na frekvenčné filtre. ....	39
Obr. 5.20: Doplnkové príklady. ....	39
Obr. 5.21: Nahratá aplikácia na MATLAB File Exchange. ....	40
Obr. 5.22: Technický problém GUI. ....	40

## ZOZNAM SKRATIEK

Skratka	Anglický význam	Slovenský význam
<b>GUI</b>	Grafical User Interface	Grafické používateľské rozhranie
<b>FR</b>	–	Fourierov rad
<b>FT</b>	Fourier Transform	Fourierova transformácia
<b>IFT</b>	Inverse Fourier Transform	Inverzná Fourierova transformácia
<b>DFT</b>	Discrete Fourier Transform	Diskrétna Fourierova transformácia
<b>LTi</b>	Linear time-invariant	Lineárne časovo invariantný
<b>MIT</b>	Massachusetts Institute of Technology	–
<b>ČVUT</b>	–	České Vysoké Učení Technické
<b>PDF</b>	Portable Document Format	Formát prenosného dokumentu
<b>IDFT</b>	Inverse Discrete Fourier Transform	Inverzná diskretná Fourierova transformácia
<b>HTML</b>	HyperText Markup Language	Hypertextový značkový jazyk
<b>CSS</b>	Cascading Style Sheets	Kaskádové štýly
<b>PP</b>	–	Pseudonáhodná postupnosť
<b>BER</b>	Bit Error Rate	Bitová chybovosť
<b>SNR</b>	Signal to Noise Ratio	Odstup signálu od šumu
<b>RAM</b>	Random-Access Memory	Pamäť s priamym prístupom

## ZOZNAM SYMBOLOV

Symbol	Význam symbolu
$t$	Spojité časová premenná
$T$	Periódá
$f(t)$	Spojité signál v časovej oblasti
$\omega$	Uhlová frekvencia
$F(\omega)$	Spojité signál vo frekvenčnej oblasti
$e$	Eulerovo číslo
$a_0$	Jednosmerná zložka
$a_n$	Koeficient kosínusových členov
$b_n$	Koeficient sínusových členov
$j$	Komplexná jednotka
$f_{vz}$	Vzorkovacia frekvencia
$f_m$	Maximálna frekvencia signálu
$n$	Číslo vzorky
$k$	Číslo príslušnej harmonickej zložky
$N$	Počet vzoriek
$x(n)$	Diskrétny signál v časovej oblasti
$X(k)$	Diskrétny signál vo frekvenčnej oblasti
$\tau$	Spojité časové posunutie
$R(\tau)$	Konvolúčna funkcia
$K(\tau)$	Korelačná funkcia
$y(\tau)$	Výstupný spojité signál
$x(t)$	Vstupný spojité signál
$h(t)$	Impulzová odozva systému

# ÚVOD

Každý deň sa stretávame so signálmi, pričom si ich často ani neuvedomujeme. Zvukové signály, ktoré počujeme okolo seba, obrazové informácie, ktoré sledujeme na obrazovkách, dokonca aj elektrické a elektromagnetické signály, ktoré umožňujú prenos informácií cez internet, predstavujú len niektoré príklady signálov s ktorými prichádzame denne do kontaktu. Signál ako fyzikálna veličina predstavuje základný prostriedok pre prenos informácie. Každý signál musí existovať v určitom systéme, ktorým môže byť napríklad prenosové médium, v ktorom sa šíri, alebo elektrický obvod, kde môže meniť svoju podobu na iný typ signálu, alebo sa zaznamenávať pre ďalšie spracovanie.

Znalosti o signáloch a systémoch sú preto kľúčové v rôznych technických a vedeckých odboroch od elektrotechniky cez telekomunikácie až po biomedicínu. Pochopenie problematiky signálov a systémov, môže ale byť pre niekoho komplikované, a preto cieľom tejto bakalárskej práce je vytvoriť GUI, ktoré bude mať formu kurzu a umožní moderným a interaktívnym spôsobom uľahčiť proces učenia a pochopenia týchto zložitých konceptov. Vytvorené GUI bude slúžiť, ako výučbová pomôcka pre predmet Úvod do signálov a systémov.

Teoretická časť práce začína jednou kapitolou, ktorá sa venuje úvodu do teórie signálov a systémov. V tejto časti sú opísané definície, matematické vzťahy a rozdelenie signálov a systémov. Ďalej nasleduje kapitola s prehľadom dostupných bezplatných kurzov, ktoré sa zameriavajú na problematiku signálov a systémov. Posledná kapitola v teoretickej časti je zameraná na programovacie jazyky, ktoré umožňujú spracovať a vizualizovať signály a systémy.

V praktickej časti práce je jedna kapitola venovaná vysvetleniu procesu návrhu a tvorby jednotlivých častí GUI. V tejto časti sa podrobne zaoberáme metodikou a postupmi, ktoré sme použili pri vytváraní GUI, vrátane výberu vhodných nástrojov, technológií a implementácie jednotlivých komponentov GUI. Po tejto kapitole nasleduje ďalšia kapitola, ktorá je venovaná popisu a princípu činnosti vytvoreného GUI. Tu detailne popisujeme funkcie a možnosti, ktoré naše GUI poskytuje užívateľom, ako aj jeho štruktúru a spôsob použitia. Súčasťou tejto kapitoly sú aj technické problémy vytvoreného GUI a porovnanie nami vytvoreného kurzu s existujúcimi riešeniami.

# 1 TEORETICKÝ ÚVOD DO SIGNÁLOV A SYSTÉMOV

Pojem signál vyjadruje určitú fyzikálnu veličinu nesúcu informáciu, ktorá mení svoje fyzikálne charakteristiky s časom. Signál je vždy viazaný na nejaký systém (sústavu), v ktorom sa šíri, zaznamenáva alebo transformuje na iný typ signálu [1,2].

## 1.1 Základné rozdelenie signálov

Signály je možné rozdeliť podľa viacerých kritérií. Podľa toho ako sú definované hodnoty, delíme signály na:

- **deterministické signály** – je ich možné opísať deterministickou funkciou času a vopred určiť hodnotu v akomkoľvek časovom okamihu,
- **stochastické signály** – nemôžeme opísať známou funkciou času a vopred určiť ich hodnotu v čase, sú reprezentované súborom štatistických parametrov [1,3].

Ďalším kritériom je povaha definičného oboru a obor hodnôt, podľa ktorého rozdeľujeme signály na:

- **signály spojité v čase** – modelom je spojitá funkcia, ktorá nadobúda nekonečný počet hodnôt v čase,
- **signály diskrétne v čase** – sú vyjadrené konečným a usporiadaným súborom hodnôt v čase [1,3].

## 1.2 Periodické signály v časovej a frekvenčnej oblasti

Periodické signály patria do skupiny deterministických spojitých signálov, ktoré sa opakujú s určitou periódou na celom časovom intervale. Model periodického signálu musí vyhovovať podmienke:

$$f(t) = f(t + T) \text{ pre } t \in (-\infty, \infty) \quad (1.1)$$

kde  $T$  je číslo, vyjadrujúce periódou a  $t$  je spojitá časová premenná [2,3].

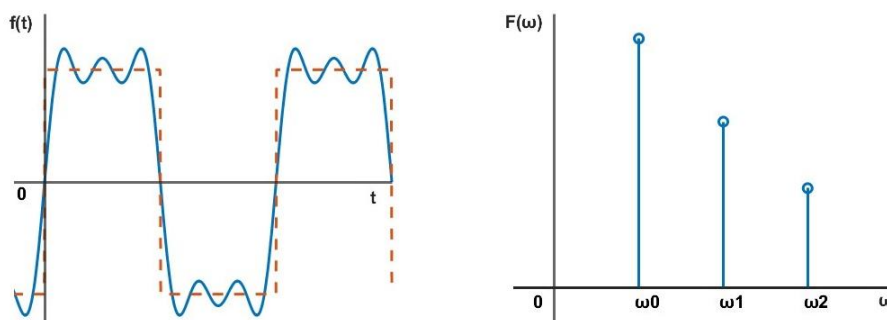
Periodické signály možno rozdeliť na harmonické a neharmonické. Harmonické signály je možné opísať sínusovou alebo kosínusovou funkciou času. Neharmonické signály, vznikajú súčtom viacerých harmonických funkcií pomocou Fourierovho radu (FR) [2,3].

FR je matematický nástroj, ktorý umožňuje vyjadriť každú periodickú funkciu, ktorá spĺňa Dirichletove podmienky, ako súčet viacerých sínusových a kosínusových funkcií. FR je možné vyjadriť v troch hlavných tvaroch, pričom najčastejšie sa používa trigonometrický tvar:

$$f(t) = a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega t) + b_n \sin(n\omega t), \quad (1.2)$$

kde  $a_0$  je časová stredná hodnota (jednosmerná zložka),  $a_n$  je koeficient kosínusových členov, a  $b_n$  je koeficient sínusových členov FR [2].

Periodické signály sú charakteristické tým, že majú diskrétné (čiarové) spektrum. Spektrum periodického signálu môžeme vyjadriť FR, kde každá frekvencia harmonickej zložky FR zodpovedá jednej spektrálnej zložke (čiare). Spektrum je špecifické tým, že má klesajúcu tendenciu smerom k vyšším harmonickým zložkám [1,3].



Obr. 1.1: Aproximácia periodického obdĺžnikového signálu: a) časová oblasť, b) frekvenčná oblasť.

### 1.3 Neperiodické signály v časovej a frekvenčnej oblasti

Neperiodické signály tvoria podskupinu deterministických signálov, ktoré nevyhovujú podmienkam pre periodické signály (vzťah 1.1). Model neperiodických signálov je reprezentovaný funkciou, ktorá je definovaná iba v určitej časti časového intervalu a mimo tohto intervalu má nulovú hodnotu [2].

Sú vo frekvenčnej doméne matematicky reprezentované ako spojité a frekvenčne neohraničené funkcie. Pre opis vlastností neperiodických signálov vo frekvenčnej oblasti využívame Fourierovu transformáciu (FT). Využitím FT získame z funkcie časového priebehu  $f(t)$  funkciu spektrálnej hustoty  $F(\omega)$ . FT je daná vzťahom:

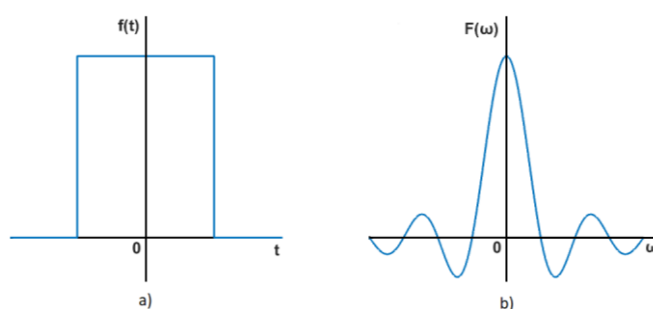
$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt, \quad (1.3)$$

kde  $F(\omega)$  je funkcia spektrálnej hustoty,  $f(t)$  je funkcia časového priebehu a  $\omega$  je uhlová frekvencia signálu [2,5].

Inverzný proces, kedy získame z funkcie spektrálnej hustoty  $F(\omega)$ , funkciu časového priebehu  $f(t)$  nazývame inverzná Fourierova transformácia (IFT). IFT je definovaná:

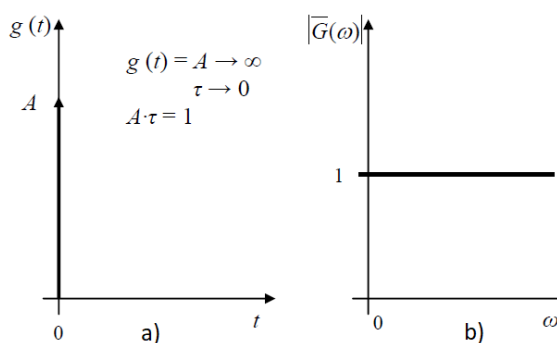
$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{-j\omega t} d\omega, \quad (1.4)$$

kde  $f(t)$  je funkcia časového priebehu,  $F(\omega)$  je funkcia spektrálnej hustoty signálu a  $\omega$  je uhlová frekvencia signálu [2,5].



Obr. 1.2: obdĺžnikový neperiodický signál: a) v časovej oblasti, b) vo frekvenčnej oblasti.

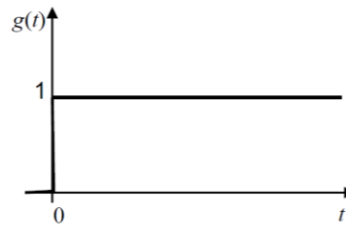
Medzi základné typy neperiodických signálov radíme Diracov impulz, ktorý je definovaný len pre  $t = 0$  a pre  $t \neq 0$  má nulovú hodnotu. Diracov impulz je špecifický tým, že jeho šírka je nekonečne úzka a amplitúda je nekonečne veľká, z čoho vyplýva, že má vzorkovaciu vlastnosť. Spektrum Diracovho impulzu je konštantné a má nekonečnú šírku [1,2,5].



Obr. 1.3: Diracov impulz: a) v časovej oblasti, b) vo frekvenčnej oblasti [1].

Ďalším základným typom je Heavisedov skok, v mnohých literatúrach označovaný aj ako jednotkový skok. Heavisedov skok vyjadruje funkciu jednoduchých nespojitostí a využíva sa na zápis kauzálnych funkcií [2].





Obr. 1.4: Heavisideov skok [1].

## 1.4 Vzorkovanie

Vzorkovanie predstavuje proces, pri ktorom sa periodicky s frekvenciou  $f_{vz}$  zo spojitého signálu pomocou postupnosti impulzov odoberajú vzorky v určitých časových okamihoch. Pri vzorkovaní musí byť dodržaná Shannon-Nyquistová podmienka:

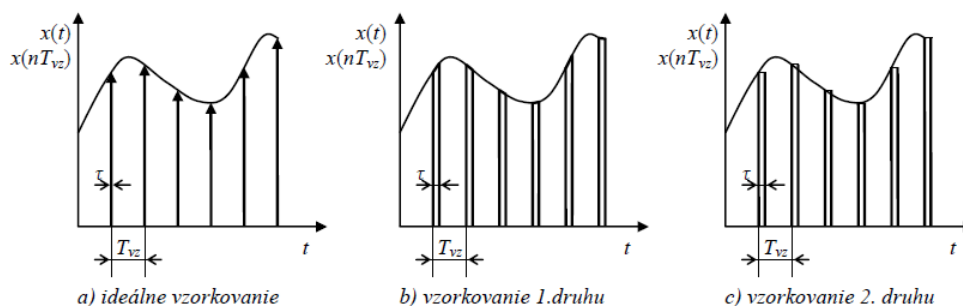
$$f_{vz} \geq 2f_m, \quad (1.5)$$

kde  $f_{vz}$  je vzorkovacia frekvencia a  $f_m$  je maximálna frekvencia signálu [1].

Pokiaľ je splnená Shannon-Nyquistová podmienka nedochádza k strate informácie a výsledná postupnosť vzoriek obsahuje pôvodnú informáciu reprezentovanú spojitým signálom [1].

Vzorkovanie je možné rozdeliť podľa tvaru vzorkovacieho impulzu na:

- **ideálne vzorkovanie** – šírka impulzu je nekonečne malá a amplitúda nekonečne veľká, jedná sa o postupnosť modulovaných Diracových impulzov (obr. 1.5a),
- **vzorkovanie 1. druhu** – šírka impulzu má konečnú hodnotu a jednotlivé impulzy kopírujú priebeh signálu (obr. 1.5b),
- **vzorkovanie 2. druhu** – šírka impulzu má konečnú hodnotu a jednotlivé impulzy sú udržiavané na konštantnej hodnote spojitého priebehu (obr. 1.5c) [1,2].

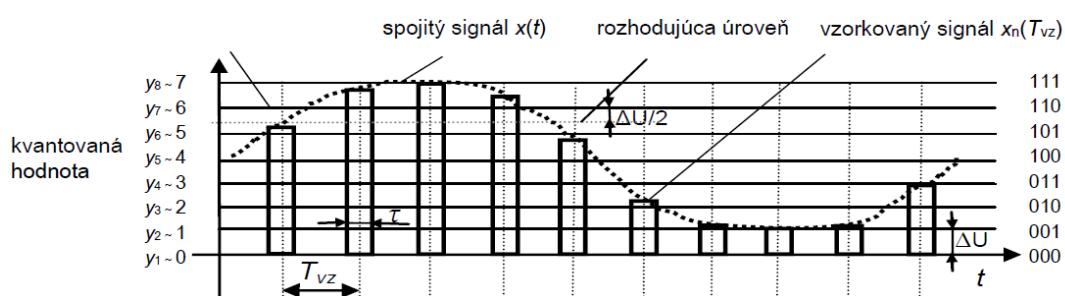


Obr. 1.5: Druhy vzorkovania [1].

## 1.5 Kvantovanie

Kvantovanie je proces, pri ktorom sa spojitý signál, ktorý môže nadobúdať nekonečný počet hodnôt, transformuje na diskretný formát, v ktorom môže nadobúdať len špecifické, diskkrétne hodnoty. Pri kvantovaní sa celkový rozsah možných veľkostí vzoriek rozdelí na určitý konečný počet kvantizačných úrovní. Každá kvantizačná úroveň má vymedzenú rozhodovaciu úroveň, ktorá určuje, či vzorka ešte spadá do danej kvantizačnej úrovne [2,6].

Kvantovaním signálu vzniká skreslenie pôvodného signálu. Toto skreslenie má charakter šumu a nazýva sa kvantizačný šum [2,6].



Obr. 1.6: Proces kvantovania [1].

## 1.6 Diskrétné signály v časovej a frekvenčnej oblasti

Diskretný signál vzniká procesom vzorkovania spojitého signálu a predstavuje konečnú usporiadanú množinu hodnôt. Matematický model diskretného signálu môžeme vyjadriť:

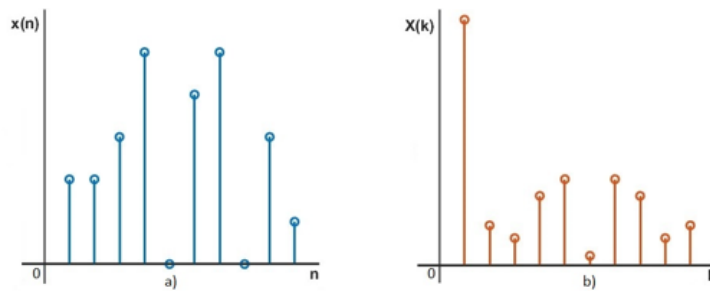
$$x[n] = \{x(0), x(1), \dots, x(k)\}, \quad (1.6)$$

kde  $x[n]$  je celková postupnosť vzoriek a  $x(k)$  predstavuje  $k$ -tu vzorku postupnosti [1,2].

Frekvenčné spektrum diskretných signálov je tiež diskrétné a pre výpočet frekvenčného spektra diskretných signálov využívame diskretnú Fourierovú transformáciu (DFT):

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j \frac{2\pi}{N} nk}, \quad (1.7)$$

kde  $X(k)$  je spektrum signálu  $x(n)$ ,  $n$  je poradové číslo vzorky,  $k$  je poradové číslo príslušnej harmonickej zložky a  $N$  je počet vzoriek [1].



Obr. 1.7: Diskrétny signál: a) v časovej oblasti, b) vo frekvenčnej oblasti.

## 1.7 Konvolúcia a korelácia

Konvolúcia a korelácia sú si matematicky veľmi podobné, ale fyzikálne sa od seba líšia. Pri analýze signálov sa často využíva pojem korelácia a pri systémoch (sústavách) je to pojem konvolúcia [3].

Konvolúcia je matematická operácia, ktorá kombinuje dve funkcie a opisuje ich prekryvanie. Týmto procesom sa vytvorí nová funkcia, ktorá predstavuje spôsob, akým na seba vzájomne pôsobia dve pôvodné funkcie. S využitím konvolúcie vieme určiť výstupný signál na základe vstupného signálu a impulzovej odozvy systému (sústavy). Konvolúciu je možné vyjadriť:

$$R(\tau) = \int_{-\infty}^{\infty} f_1(t)f_2(\tau - t)dt, \quad (1.8)$$

kde  $f_1(t)$ ,  $f_2(t)$  sú vstupné funkcie a  $\tau$  je spojité časové posunutie [3,7].

Korelácia vyjadruje určitú mieru zhodnosti, súvislosti alebo podobnosti dvoch signálov. Pokiaľ vykonávame koreláciu signálu so samým sebou oneskorením o čas  $\tau$  nazývame to autokorelačná funkcia. Keď opisujeme vzájomnú súvislosť dvoch rôznych signálov, označujeme to ako krížová korelačná funkcia. Koreláciu je možné vyjadriť:

$$K(\tau) = \int_{-\infty}^{\infty} f_1(t)f_2(t - \tau)dt, \quad (1.9)$$

kde  $f_1(t)$ ,  $f_2(t)$  sú vstupné funkcie a  $\tau$  je spojité časové posunutie [1-3].

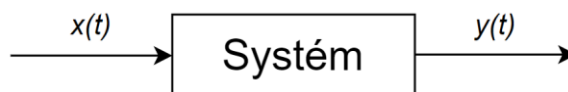
## 1.8 Stochastické signály

Stochastické signály nie je možné popísať známou matematickou funkciou času, pretože ich signálové parametre sa s časom nepredvídateľne menia. Preto pre opis stochastických signálov využívame súbor štatistických parametrov, medzi ktoré patrí hustota rozdelenia pravdepodobnosti, stredná hodnota, rozptyl a smerodajná odchýlka [1,2].

Stochastické signály môžeme rozdeliť na stacionárne, nestacionárne a ergodické. Medzi stochastické signály radíme aj šumy. Šumy predstavujú nežiaduce signály, ktoré spôsobujú rušenie pri spracovaní a prenose užitočných signálov [5].

## 1.9 Systémy

Existencia signálu je vždy viazaná na nejaký systém. Systém predstavuje množinu na seba viazaných prvkov, ktoré pod vplyvom vonkajších a vnútorných signálov vzájomne na seba pôsobia za účelom splnenia požadovanej funkcie. Charakteristickým znakom každého systému je, že má vstup a výstup. Na vstup systému pôsobí vstupný signál  $x(t)$  a na výstupe získavame výstupný signál  $y(t)$ . Výstupný signál  $y(t)$  závisí od vstupného signálu  $x(t)$  a prenosovej funkcie systému [1,2].



Obr. 1.8: Bloková schéma systému.

Systémy môžeme rozdeliť na základe matematického modelu na lineárne a nelineárne systémy a podľa časovej závislosti môžeme systémy rozdeliť na časovo invariantné (nepremenné) a časovo variantné (premenné) [2].

Systémy tiež môžeme deliť na bezpamäťové a pamäťové. Pri bezpamäťových systémoch výstupná hodnota signálu závisí len od hodnoty vstupného signálu v danom okamihu. Pamäťové systémy využívajú akumulčný prvok a ich výstupná hodnota nezávisí len od vstupnej hodnoty signálu v danom okamihu, ale aj od predchádzajúcich hodnôt [2].

Špecifickým prípadom je lineárne časovo invariantný (LTI) systém, ktorý je súčasne lineárny aj časovo invariantný. LTI systém je charakterizovaný impulzovou odozvou. Impulzová odozva vyjadruje odozvu LTI systému na Diracov impulz. Pokiaľ poznáme vstupný signál a impulzovú odozvu LTI systému, vieme s využitím konvolúcie zistiť výstupný signál:

$$y(\tau) = \int_{-\infty}^{\infty} x(t)h(t - \tau)dt, \quad (1.10)$$

kde  $y(\tau)$  je výstupný signál,  $x(t)$  je vstupný signál a  $h(t)$  je impulzová odozva LTI systému [8,9].

## 2 DOSTUPNÉ KURZY A APLIKÁCIE

V súčasnosti existuje viacero kurzov a aplikácii, ktoré sa venujú problematike signálov a systémov. Niektoré kurzy a aplikácie ponúkajú interaktívne učebné prostredie, kde sa užívatelia môžu učiť základné a pokročilé koncepty prostredníctvom videí, cvičení a príkladov. Niektoré kurzy sú dostupné bezplatne, avšak značná časť kurzov je spoplatnená a cena kurzu sa môže v niektorých prípadoch pohybovať až v stovkách eur. V tejto kapitole sa preto budeme venovať len bezplatným kurzom. Výhodou spoplatnených kurzov je, že často poskytujú okrem študijných materiálov aj odborného inštruktora. Taktiež je dôležité poznamenať, že okrem jedného kurzu v českom jazyku sú všetky kurzy dostupné iba v anglickom jazyku.

### 2.1 Kurz od MIT

Americká univerzita Massachusetts Institute of Technology (MIT) poskytuje bezplatný kurz venujúci sa problematike signálov a systémov. Kurz obsahuje 26 video prednášok profesora Alana V. Oppenheima [10].

Prvé prednášky a materiály sa venujú parametrom signálu a základnému rozdeleniu signálov a systémov. Po základnom rozdelení signálov a systémov nasledujú prednášky, konkrétne zamerané na témy zaoberajúce sa konvolúciou, koreláciou, FR, FT, DFT a LTI systémami. Kurz taktiež rieši témy, ktoré sú nad rámec predmetu Úvod do signálov a systémov vyučovaného u nás na univerzite [10].

Výhodou kurzu je, že ku každej prednášanej téme poskytuje výpočtové príklady s riešením a aj poznámky vytvorené prednášajúcim [10].

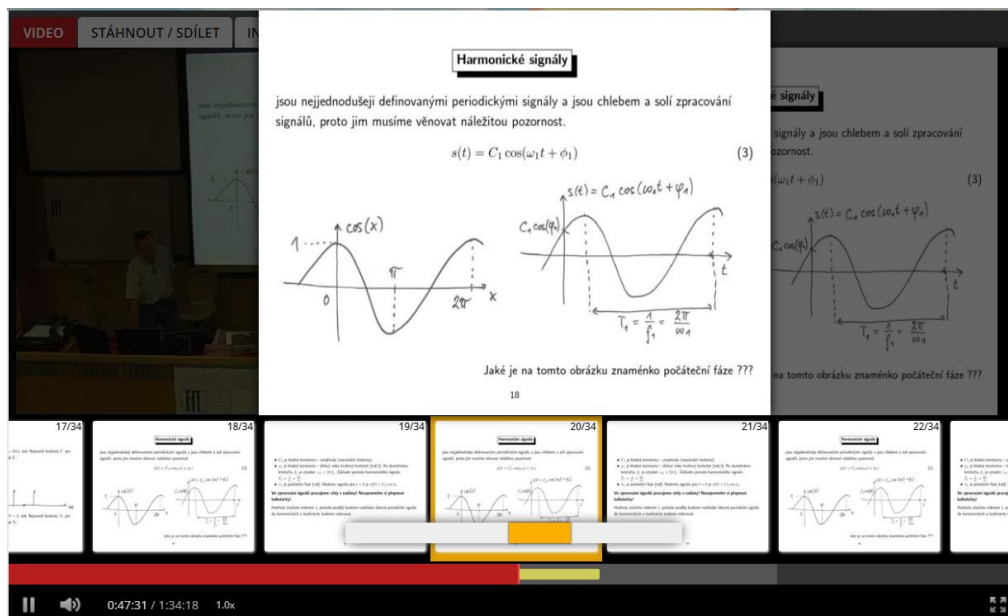
Nevýhodou kurzu je, že neobsahuje žiadne teoretické testy pre overenie nadobúdaných vedomostí a kurz je cez 10 rokov starý, čomu zodpovedá aj kvalita video nahrávok. Ďalšou nevýhodou je, že sa nezaobrá problematikou stochastických signálov [10].



Obr. 2.1: Kurz od MIT [10].

## 2.2 Kurz od ČVUT

Kurz z české univerzity České Vysoké Učení Technické (ČVUT) je bezplatný kurz dostupný v českém jazyku. Obsahem kurzu je 13 video přednášek, přednášaných profesorem Jánem Černockým. Prednášky sa venujú základným typom signálov a systémov, frekvenčným spektrám signálov, konvolúcii, vzorkovaniu a celková osnova kurzu približne koreluje s osnovou predmetu Úvod do signálov a systémov. Kurz umožňuje počas sledovania video prednášky zobrazit' jednotlivé časti prezentácie (obr. 2.2) a stiahnuť aj celú prezentáciu vo formáte PDF [11].



Obr. 2.2: Kurz od ČVUT – možnosť prezerania jednotlivých častí prezentácie [11].

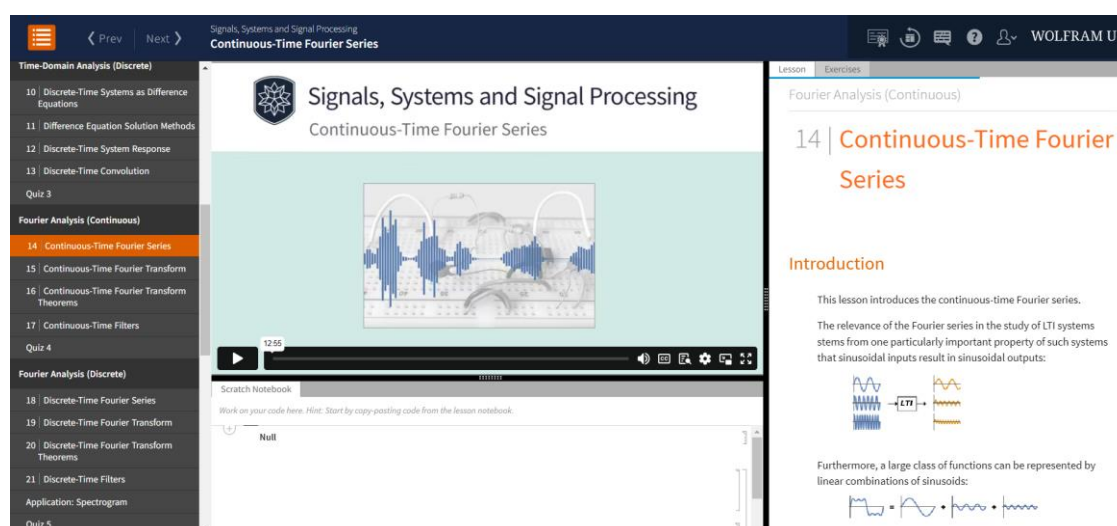
V niektorých prednáškach sú zahrnuté aj výpočtové príklady na precvičenie, ale okrem toho kurz neobsahuje žiadne iné doplnkové študijné materiály, testy a príklady. Kurz nie je veľmi aktuálny a je niekoľko rokov starý, takže kvalita videa a zvuku nie je najlepšia [11].

## 2.3 Kurz od Wolfram

Spoločnosť Wolfram poskytuje viacero bezplatných kurzov z odvetia ekonomiky, matematiky, fyziky a iných prírodných vied. Kurzy sú umiestnené na Wolfram Cloud a umožňujú interaktívne skúmať koncepty pomocou funkcií jazyka Wolfram. Kurz od Wolframu je jediným kurzom z nášho zoznamu, ktorý si vyžaduje registráciu [12].

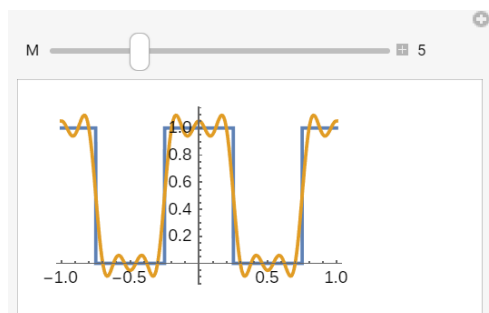
Obsah kurzu je rozdelený do 8 kapitol, ktoré sa venujú základnému rozdeleniu signálov a systémov, analýze spojitých a diskretných signálov, spojitej a diskkrétnej Fourierovej analýze (FR, FT, DFT), Laplacovej a Z transformácii, vzorkovaniu a číslicovým filtrom. Jednotlivé kapitoly sú rozdelené do 4 až 5 podkapitol, zameraných na konkrétnu tému z daného okruhu. V každej podkapitole je video, vysvetľujúce preberanú látku a napravo od videa sa nachádza študijný textový materiál, zobrazený na obr. 2.3 [13].

Pod videom sa nachádza textové pole, v ktorom je možné si písať poznámky a Wolfram kód. Poznámky a kód je možné si následne poslať na email, uložiť ako Wolfram notebook alebo vo forme obrázku [13].



Obr. 2.3: Užívateľské rozhranie kurzu Wolfram [13].

V študijných materiáloch sa nachádzajú aj vypočítané príklady a interaktívne ukážky, kde užívateľ môže meniť parametre na základe, ktorých sa mení priebeh zobrazeného signálu. Príklad interaktívnej ukážky je zobrazený na obr. 2.4 [13].



Obr. 2.4: Interaktívna ukážka – aproximácia obdĺžnikového signálu [13].

Obsahom jednotlivých podkapitol sú aj príklady na precvičenie, ktoré po kliknutí na tlačidlo *solution* ukážu riešenie. Úlohy obsiahnuté v kurze sa zameriavajú aj na spracovanie a vizualizáciu signálov prostredníctvom jazyka Wolfram [13].

Po každej kapitole je dostupný test s 10 otázkami, ktorý slúži na overenie vedomostí získaných z danej kapitoly. Testy je možné vyplniť viackrát. Testové otázky sú zväčša zamerané na teoretické vedomosti, ale v otázkach môžu byť aj výpočtové príklady. Absolvované testy a celkový progres kurzu je možné sledovať v záložke *track my progress*, kde po úspešnom absolvovaní všetkých 8 testov nad 60 % je možné bezplatne získať certifikát o absolvovaní kurzu [13].

Kurz od spoločnosti Wolfram je spomedzi bezplatne dostupných kurzov najlepšie spracovaný, pretože ako jediný poskytuje interaktívny spôsob učenia signálov a systémov.

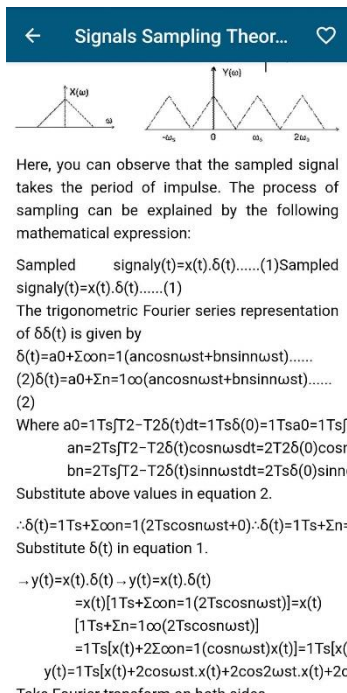
## 2.4 Android aplikácie

Aplikácie Signals and Systems a Signals Systems sú dostupné na platforme Android v Google Play Store. Obe aplikácie sú obsahovo totožné, len jednotlivé témy sú inak chronologicky usporiadané a samozrejme grafický dizajn je mierne odlišný. Obsah študijných materiálov v aplikáciách je nad rámec všetkých vyššie uvedených kurzov. Materiály sa venujú širokému spektru tém od základnej klasifikácie signálov v časovej a frekvenčnej oblasti až po základy opisu a spracovania signálov v jazyku MATLAB.

Aplikácie však trpia viacerými nedostatkami. Prvý nedostatok je zobrazený na obr. 2.6, kde je formát jednotlivých vzťahov a textov dosť neprehľadný a v niektorých



častiach až nečitateľný. Taktiež materiály nemajú uvedené žiadne zdroje, takže nie je možné určiť dôveryhodnosť materiálov. Ďalším nedostatkom aplikácií je, že sú plné reklám. Reklamy prekážajú vo viditeľnosti, pretože niektoré zaberajú takmer polovicu displeja, zobrazené na obr. 2.7. Posledným nedostatkom je, že aplikácie neobsahujú žiadne videá, príklady, cvičenia a ani interaktívne ukážky, sú to čisto len textové materiály.



← Signals Sampling Theor... ♥

Here, you can observe that the sampled signal takes the period of impulse. The process of sampling can be explained by the following mathematical expression:

Sampled signal  $y(t) = x(t) \cdot \delta(t) \dots (1)$

Sampled signal  $y(t) = x(t) \cdot \delta(t) \dots (1)$

The trigonometric Fourier series representation of  $\delta(t)$  is given by

$\delta(t) = a_0 + \sum_{n=1}^{\infty} (a_n \cos n\omega t + b_n \sin n\omega t) \dots$

(2)  $\delta(t) = a_0 + \sum_{n=1}^{\infty} (a_n \cos n\omega t + b_n \sin n\omega t) \dots$

(2)

Where  $a_0 = \frac{1}{T} \int_{-T/2}^{T/2} \delta(t) dt = \frac{1}{T} \delta(t) = \frac{1}{T}$

$a_n = \frac{2}{T} \int_{-T/2}^{T/2} \delta(t) \cos n\omega t dt = \frac{2}{T} \delta(t) \cos n\omega t = \frac{2}{T} \cos n\omega t$

$b_n = \frac{2}{T} \int_{-T/2}^{T/2} \delta(t) \sin n\omega t dt = \frac{2}{T} \delta(t) \sin n\omega t = \frac{2}{T} \sin n\omega t$

Substitute above values in equation 2.

$\therefore \delta(t) = \frac{1}{T} + \sum_{n=1}^{\infty} (2 \cos n\omega t) \therefore \delta(t) = \frac{1}{T} + \sum_{n=1}^{\infty} 2 \cos n\omega t$

Substitute  $\delta(t)$  in equation 1.

$\rightarrow y(t) = x(t) \cdot \delta(t) \rightarrow y(t) = x(t) \cdot \left[ \frac{1}{T} + \sum_{n=1}^{\infty} 2 \cos n\omega t \right] = x(t) \left[ \frac{1}{T} + \sum_{n=1}^{\infty} 2 \cos n\omega t \right]$

$= \frac{1}{T} x(t) + 2 \sum_{n=1}^{\infty} \cos n\omega t x(t) = \frac{1}{T} x(t) + 2 \sum_{n=1}^{\infty} \cos n\omega t x(t)$

$y(t) = \frac{1}{T} x(t) + 2 \sum_{n=1}^{\infty} \cos n\omega t x(t)$

Take Fourier transform on both sides

Obr. 2.5: Formát rovníc a textu v aplikáciách.

**Introduction:**

In engineering, we often deal with the idea of an action occurring at a point. Whether it be a force at a point in space or a signal at a point in time, it becomes worth while to develop some way of quantitatively defining this. This leads us to the idea of a unit impulse, probably the second most important function, next to the complex exponential, in systems and signals course.

**The impulse function:**

**Dirac Delta Function:**

The Dirac Delta function, often referred to as the unit impulse or delta function, is the function that defines the idea of a unit impulse.

This function is one that is infinitesimally narrow, infinitely tall, yet integrates to unity.

Perhaps the simplest way to visualize this is as

Ad

Temu: Nakupujte jako miliardář

★★★★★

Přijďte a nakupujte na Temu. Stáhněte si aplikaci Temu a nakupujte jako miliardář. Doprava zdarma pro nové hosty.

TEMU CATEGORIES

INŠTALOVAT

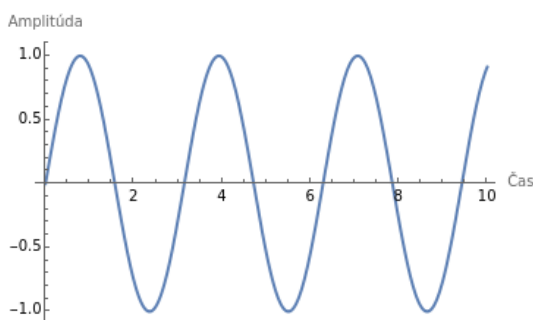
Obr. 2.6: Reklama v aplikáciách.

### 3 PROGRAMOVACIE JAZYKY PRE SPRACOVANIE A OPIS SIGNÁLOV

Z hľadiska súčasných a budúcich výziev pri spracovaní a opise signálov sú dostupné viaceré komerčné ale aj bezplatné programovacie jazyky. Niektoré jazyky majú už priamo v jadre implementované funkcie pre spracovanie a opis signálov. Pri niektorých sú dostupné viaceré externé knižnice, ktoré poskytujú tieto funkcie. V tejto kapitole sa budeme venovať niektorým komerčným ale aj bezplatným programovacím jazykom.

#### 3.1 Jazyk Wolfram

Jazyk Wolfram je symbolický jazyk, ktorý bol navrhnutý v takej šírke a jednotnosti, aby bolo možné rýchlo vyvíjať výkonné skripty. Jazyk Wolfram priamo v jadre disponuje viacerými funkciami na analýzu, spracovanie a vizualizáciu signálov v časovej a frekvenčnej oblasti. Funkcie zahŕňujú napríklad generovanie základných typov signálov, funkcie pre výpočet FR, FT, DFT, konvolúcie, korelácie a pokročilejšie funkcie na návrh digitálnych a analógových filtrov [14-16].



Obr. 3.1: Sinusový signál v jazyku Wolfram.

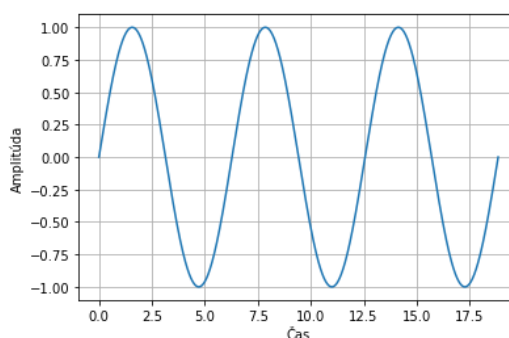
Wolfram je komerčným programovacím jazykom avšak, poskytuje bezplatný prístup prostredníctvom ich *cloud*. Tento prístup k jazyku sa využíva aj v praktických príkladoch, ktoré sú súčasťou jednotlivých kurzov od tejto spoločnosti. Pre získanie plnohodnotného vývojového prostredia pre jazyk Wolfram, ktoré sa nazýva Wolfram notebook si je potrebné túto službu zakúpiť [14].

#### 3.2 Jazyk Python

Python je bezplatný, interpretovaný a objektovo orientovaný programovací jazyk, ktorý má jednoduchú a veľmi ľahko čitateľnú syntax. Sám o sebe Python neobsahuje v jadre

žiadne funkcie a metódy pre opis a spracovanie signálov. Pre Python ale existujú viaceré externé knižnice, ktoré tento problém riešia [17,18].

V oblasti opisu a spracovania signálov patria medzi najpoužívanejšie knižnice SciPy, NumPy a Matplotlib. Knižnice NumPy a SciPy poskytujú široké spektrum funkcií. Medzi ktoré patria napríklad funkcie pre výpočet FT, DFT, FR, konvolúcie, korelácie, filtrov a lineárnych systémov. Okrem týchto funkcií obsahujú aj nástroje na generovanie deterministických a náhodných signálov. Knižnica Matplotlib slúži na vizualizáciu dát vypočítaných pomocou knižníc NumPy alebo SciPy. Vhodné je používať tieto knižnice v kombinácii, pretože každá z nich poskytuje špecifické funkcie, ktoré dopĺňajú a rozširujú možnosti druhej [19-21].



Obr. 3.2: Sínusový signál vytvorený knižnicami NumPy a Matplotlib v jazyku Python.

Pre Python existuje aj viacero knižníc na vytvorenie grafického užívateľského rozhrania (GUI). Medzi najviac používané patria knižnice Tkinter, PyGame, Kivy a PyQt. Nevýhodou týchto knižníc je, že neobsahujú žiaden grafický nástroj pre rýchlejší návrh a dizajn GUI, a preto je potrebné GUI programovať riadok po riadku [22].

### 3.3 Jazyk MATLAB

MATLAB je komerčný programovací jazyk vytvorený spoločnosťou MathWorks určený pre inžinierske a vedecké výpočty, simulácie, modelovanie, vizualizáciu dát a tvorbu aplikácií. MATLAB je možné rozšíriť o rôzne knižnice, ktoré sa v MATLAB nazývajú *Toolbox*. Tieto *Toolbox* sú vytvorené priamo spoločnosťou MathWorks, alebo tretími stranami [23,24].

MATLAB priamo jadre poskytuje širokú paletu funkcií pre analýzu a manipuláciu so signálmi. Tieto funkcie zahŕňajú generovanie deterministických a náhodných signálov, analýzu systémov a signálov v časovej a v frekvenčnej oblasti prostredníctvom matematických funkcií konvolúcia, korelácia, FT, IFT, DFT. V kontexte signálov

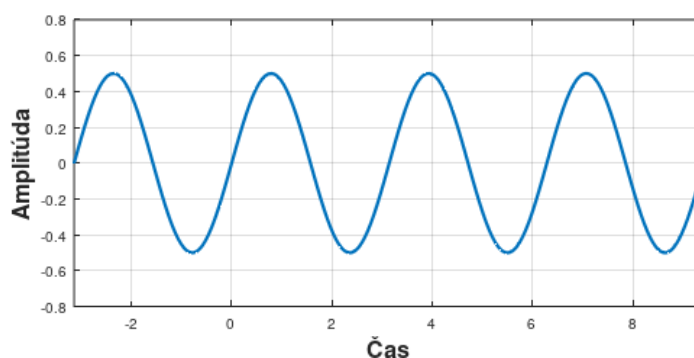
a systémov existujú aj viaceré *Toolbox*, ktoré poskytujú ešte účinnejšie nástroje pre prácu so signálmi a systémami [25].

V rámci MATLAB je k dispozícii grafický nástroj App Designer, ktorý uľahčuje a urýchľuje proces programovania GUI. App Designer integruje dve hlavné zložky tvorby aplikácií, ktorými sú rozloženie vizuálnych komponentov a programovanie logiky aplikácie. Tento nástroj umožňuje rýchly prechod medzi vizuálnym návrhom a úpravou zdrojového kódu [26].

### 3.4 Jazyk GNU Octave

GNU Octave alebo iba Octave je vysoko úrovňový bezplatný programovací jazyk primárne určený pre numerické výpočty. Je čiastočne kompatibilný s MATLAB a považuje sa za bezplatnú alternatívu k MATLAB. Oproti MATLAB má však Octave mierne odlišnú syntax [27,28].

V súvislosti so signálmi a systémami poskytuje Octave takmer totožné nástroje ako MATLAB. Rozdielom je, že jednotlivé funkcie majú oproti MATLAB odlišné názvy. Nástroje jazyka Octave je taktiež možné rozšíriť prostredníctvom externých knižníc. V oblasti signálov a systémov Octave disponuje knižnicou s názvom *Signal*, avšak možnosti tejto knižnice sú obmedzené v porovnaní s nástrojmi dostupnými v MATLAB *Toolbox* [29,30].



Obr. 3.3: Sínusový signál v jazyku Octave.

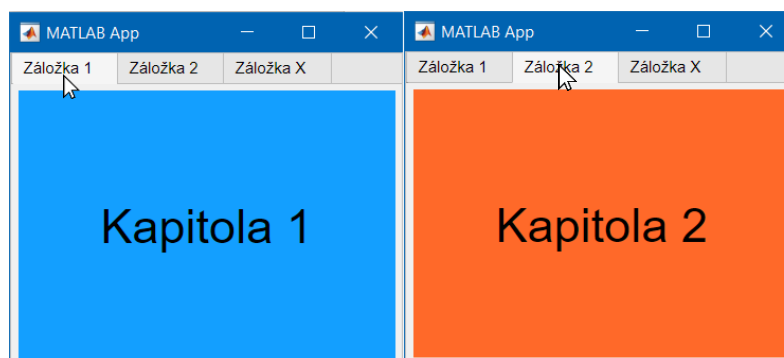
Octave má k dispozícii len obmedzené množstvo komponentov pre tvorbu GUI. Tieto komponenty zahŕňajú len jednoduché dialógové okná, ukazovatele priebehu (*progress bar*) a grafické prvky pre vykreslenie a vizualizáciu dát. Podobne ako v prípade Pythonu, aj v prípade Octave absentuje nástroj na zrýchlenie procesu vytvárania GUI [31].

## 4 NÁVRH A TVORBA GUI

Na vytvorenie GUI sme zvolili programovací jazyk MATLAB, pretože poskytuje v samotnom jadre veľké množstvo funkcií pre vizualizáciu a prácu so signálmi, preto nebolo potrebné inštalovať a importovať externé knižnice. Okrem samotného jazyku MATLAB sme pre textové časti GUI využili webové jazyky (HTML, CSS, JavaScript) a pre písanie rovníc jazyk LaTeX. Pre dizajn návrhu nebol potrebný žiadny externý softvér, akým je napríklad Figma, pretože pre tento účel sme využili priamo MATLAB App Designer, ktorý umožňuje rozložiť a prispôbiť GUI komponenty a priamo ich otestovať, bez nutnosti programovania.

### 4.1 Vizuálne členenie a štruktúra GUI

Pre vizuálne a obsahové rozdelenie GUI sme zvolili komponent *Tab Group*, ktorý nám vytvoril skupinu záložiek. Medzi jednotlivými záložkami je možné sa preklikať v samotnom vizuálnom návrhu aj v spustenej aplikácii. Vytvorené záložky je možné pomenovať a upraviť im veľkosť, farbu a font písma. Každá záložka predstavuje samostatný nezávislý kontajner, do ktorého je možné vkladať ďalšie komponenty GUI.

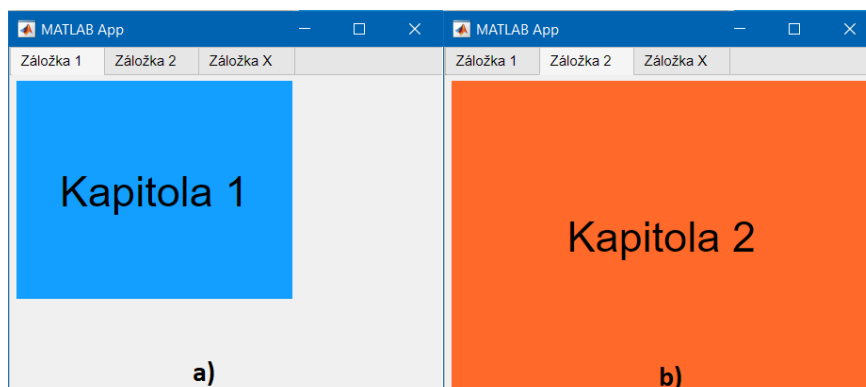


Obr. 4.1: Možnosť prekliknutia medzi záložkami.

Samotný komponent *Tab Group* sme ale predtým vložili do responzívneho rodičovského kontajnera *Grid Layout*. Tento komponent nám vytvoril mriežku s určitým počtom riadkov a stĺpcov, ktorých počet je podľa potreby možné meniť. Do vytvorených riadkov a stĺpcov je možné vkladať grafické komponenty. V tomto prípade sme do celej mriežky vložili iba komponent *Tab Group*. To nám zabezpečilo, že pri zmenšení alebo zväčšení okna aplikácie sa nám aj samotný kontajner *Tab Group* automaticky prispôbí.

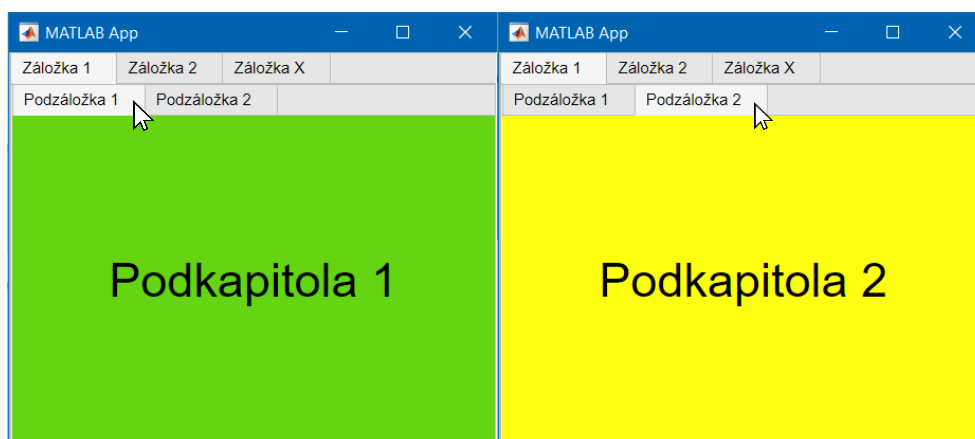
Pri vložení komponentov GUI do jednotlivých záložiek však nastal problém zobrazený na obr. 4.2. Problémom bolo, že samotný *Tab Group* bol responzívny ale

vložené komponenty už neboli responzívne a pre vyriešenie tohto problému bolo potrebné aplikovať *Grid Layout* aj do jednotlivých záložiek, do ktorých sme následne vkladali jednotlivé komponenty.



*Obr. 4.2: Komponent GUI pri zväčšení okna aplikácie a) bez využitia Grid Layout b) s využitím Grid Layout.*

Jednotlivé záložky sme následne rozdelili na viaceré časti s využitím podobného princípu ako v predošlom prípade. Do jednotlivých záložiek sme aplikovali *Grid Layout* do ktorého, sme následne vložili komponent *Tab Group*. Týmto postupom sme vytvorili záložku, ktorá je rozdelená na ďalšie nezávislé podzáložky zobrazené na obr. 4.3. Na podzáložky bolo taktiež potrebné aplikovať *Grid Layout*, aby komponenty GUI vložené do konkrétnej podzáložky boli responzívne.



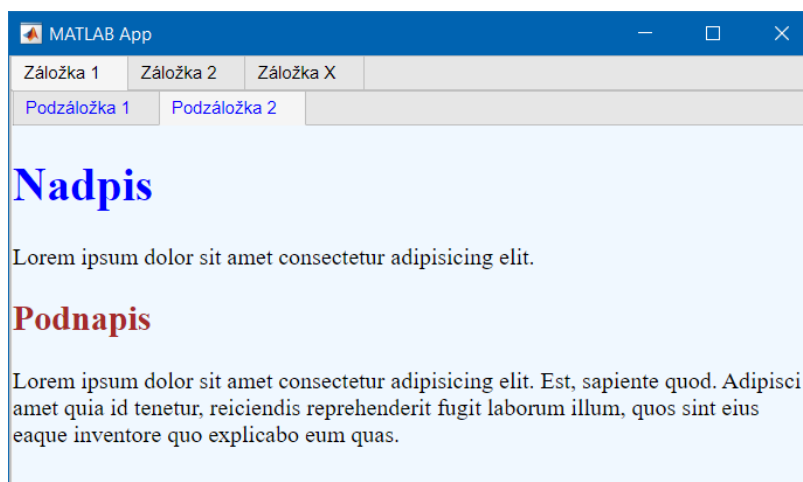
*Obr. 4.3: Princíp členenia jednotlivých kapitol na podkapitoly.*

Takto vytvorenú šablónu, kde nám záložky predstavujú kapitoly a podzáložky predstavujú podkapitoly, môžeme dopĺňať obsahom a ďalšími komponentami GUI.

Obsahovo bude jedna kapitola zameraná na jednu z tém predmetu Úvod do signálov a systémov. Do podkapitol budeme k danej téme integrovať textové učebné materiály, testy, vizuálne ukážky a príklady, ktoré budú vizualizované pomocou jazyka MATLAB.

## 4.2 Grafický návrh teoretickej textovej časti

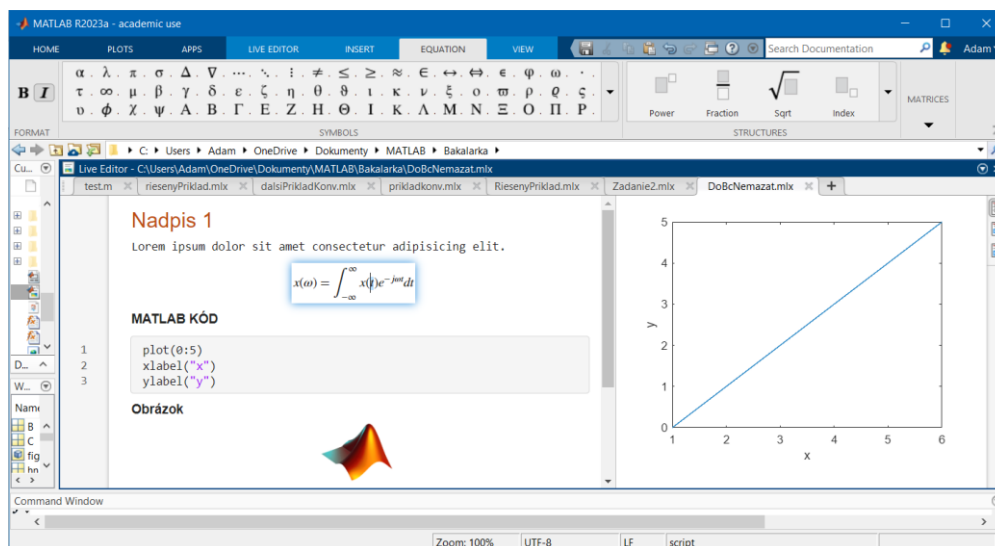
Na zobrazenie textovej časti sme využili komponent HTML, ktorý umožňuje zobraziť lokálny HTML dokument v GUI. HTML sme využili z dôvodu, že MATLAB neposkytuje žiadny grafický komponent, kde by bolo možné zobrazovať väčšie množstvo textu. Taktiež MATLAB textové komponenty neumožňujú zvýrazniť ani upraviť určitú časť textu, akými sú nadpisy, podnadvisy a iné.



Obr. 4.4: HTML komponent s HTML dokumentom.

Pre urýchlenie a uľahčenie práce sme pre vytvorenie HTML dokumentu využili jeden z MATLAB nástrojov, ktorý sa nazýva Live Script. Vo všeobecnosti Live Script poskytuje vytvorenie interaktívneho súboru, kde je možné súčasne písať text, rovnice a MATLAB kód. Napísaný kód je možné spustiť priamo v editore a výstup kódu sa nám zobrazí v textovej ploche Live Script. Live Script umožňuje zvýraznenie a členenie textu, ale jeho možnosti sú relatívne obmedzené oproti iným textovým editorom, avšak pre naše účely boli tieto možnosti dostačujúce. Okrem možnosti písania textu poskytuje Live Script vkladanie obrázkov a rovníc. Rovnice je možné jednoducho vytvoriť pomocou vstavaného editoru EQUATION zobrazeného na obr. 4.5.

Veľkou výhodou a dôvodom, prečo sme využili Live Script bolo, že vytvorený súbor je možné vyexportovať do formátu PDF, Word, LaTeX a pre nás najdôležitejší formát HTML. Využitím Live Script sme ušetrili veľké množstvo času, pretože písanie rovníc v jazyku HTML je relatívne komplikovaný a hlavne zdĺhavý proces.



Obr. 4.5: MATLAB Live Script.

### 4.3 Návrh teoretických testov

Pri návrhu testov už bolo okrem vizuálneho návrhu dôležité vytvoriť aj programovaciu logiku, ktorá určovala spôsob kontroly otázok so správnymi odpoveďami a výpočet celkového hodnotenia testu. Pre testy sme vytvorili 2 návrhy so 4 testovými otázkami, v ktorých sme použili odlišné postupy a komponenty GUI.

Prvotný návrh testových otázok bol navrhnutý s využitím komponentu *Radio Button Group*, ktorý nám vytvoril skupinu *Radio Button*, pričom jedna táto skupina predstavovala jednu otázku. Na jednej strane *Radio Button Group* zabezpečuje, že v danej skupine môže byť zakliknuté len jedno tlačidlo a má implementované metódy na získanie zakliknutého tlačidla, takže nie je nutné tieto funkcie programovať. Na druhej strane *Radio Button Group* je kontajner, ktorý neumožňuje do neho vkladať ďalšie kontajnery, to znamená, že nie je možné do neho vložiť *Grid Layout*. Toto obmedzenie spôsobilo, že celý kontajner bol responzívny, ale jednotlivé *Radio Button* neboli.

Proces vyhodnotenia testu, bol spustený tlačidlom *Submit*. Princíp kontroly odpovedí spočíval v tom, že sme za pomoci metódy *ButtonDownFcn*, ktorú má implantovanú *Radio Button Group*, získali všetky zakliknuté *Radio Button*. Následne sme vytvorili cyklus, v ktorom sme zo zakliknutých *Radio Button*, získali ich názov vo forme *string* a pomocou podmienky sme tieto hodnoty porovnali s hodnotami v poli so správnymi odpoveďami. Výsledné hodnotenie bolo vypočítané ako súčet správne zodpovedaných otázok, vydelené celkovým počtom otázok a táto hodnota bola vynásobená 100, z čoho sme



získali percentuálny výsledok testu. Výsledok bol zobrazený v komponente *Label*, ktorý sa nachádzal nad tlačidlami *Submit* a *Restart*.

Okrem tlačidla *Submit* sme vytvorili aj tlačidlo *Restart*, pomocou ktorého bolo možné vynulovať všetky zodpovedané otázky a výsledok testu a test bolo možné napísať znovu.

<p>Aký je správny zápis vzťahu pre priamu Fourierovu transformáciu?</p> <p><input type="radio"/> A <math>\int_{-\infty}^{\infty} f(\omega)e^{-j\omega t} dt</math></p> <p><input type="radio"/> B <math>\int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt</math></p> <p><input type="radio"/> C <math>\frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega)e^{j\omega t} d\omega</math></p> <p><input type="radio"/> D <math>\frac{2}{T} \int_0^T x(t) \sin(\pi \omega t) dt</math></p>	<p>Frekvenčné spektrum neperiodických signálov je</p> <p><input type="radio"/> A spojité a frekvenčne neohraničené</p> <p><input type="radio"/> B diskrétné</p> <p><input type="radio"/> C konštatné a nekonečne široké</p> <p><input type="radio"/> D harmonické</p>
<p>Aký je správny zápis vzťahu pre inverznú Fourierovu transformáciu?</p> <p><input type="radio"/> A <math>\int_{-\infty}^{\infty} f(\omega)e^{-j\omega t} dt</math></p> <p><input type="radio"/> B <math>\int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt</math></p> <p><input type="radio"/> C <math>\frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega)e^{j\omega t} d\omega</math></p> <p><input type="radio"/> D <math>\frac{2}{T} \int_0^T x(t) \sin(\pi \omega t) dt</math></p>	<p>Čím je signál(impulz) širší v časovej oblasti tým je jeho spektrum</p> <p><input type="radio"/> A užšie</p> <p><input type="radio"/> B širšie</p> <p><input type="radio"/> C rovnaké</p> <p><input type="radio"/> D spojité</p>

Obr. 4.6: Pôvodný návrh testov.

Pre odstránenie problémov s chybnou responzivitou predchádzajúceho návrhu sme v druhom návrhu zvolili komponent *Panel* ako rodičovský kontajner, do ktorého je možné aplikovať *Grid Layout*. V tomto návrhu sme vybrali ako tlačidlový komponent *Check Box*, pretože komponent *Radio Button* je možné vložiť len do rodičovského kontajneru typu *Radio Button Group*. Graficky, je teda jedna otázka zložená z kontajneru *Panel*, do ktorého sú vložené 4 *Check Box*, 4 *Label* s odpoveďami a *Label* s názvom otázky. Pre písanie vzťahov, sme komponentu *Label* zvolili interpreter LaTeX, a vkladali sme vzťahy vo forme LaTeX kódu.

Tento zvolený návrh však umožňoval zakliknúť viacero *Check Box* súčasne. Aby sme tomu zabránili, vytvorili sme funkciu, ktorá pri kliknutí na *Check Box*, skontroluje či už na danom *Panel* nie je označený iný *Check Box*. V prípade, že je iný *Check Box* už označený, funkcia zabezpečí, že pôvodne označený *Check Box* sa odznačí a označený zostane iba novozvolený *Check Box*.

Vyhodnotenie testu je aj v tomto návrhu spustené tlačidlom *Submit*. Po stlačení tlačidla, je spustená nami vytvorená funkcia, ktorá zo všetkých *Panel* vyfiltruje iba označené *Check Box*. Po získaní všetkých označených *Check Box*, sú zvyšné kroky kontroly a výpočtu totožné s predošlým návrhom s rozdielom, že výsledok testu je

v tomto návrhu zobrazený v samostatnom okne zobrazenom na obr. 4.7. Činnosť tlačidla *Restart* ostáva totožná s predošlým návrhom.

The screenshot shows a test interface with three questions (Otázka č. 1, 2, 3) and a small dialog box in the center.

**Otázka č. 1:** Aký je správny zápis vzťahu pre priamu DFT?

- ☒ A)  $\sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}nk}$
- ☐ B)  $\frac{1}{N} \sum_{n=0}^{N-1} X(k)e^{j\frac{2\pi}{N}nk}$
- ☐ C)  $\frac{1}{N} \sum_{n=0}^{N-1} X(k)e^{j\omega n}$
- ☐ D)  $\sum_{n=0}^{N-1} x(n)e^{-jkn}$

**Otázka č. 2:** Nyquistová podmienka pri vzorkovaní hovorí, že:

- ☒ A)  $f_{vz} = 2f_m$
- ☐ B)  $f_{vz} \leq 2f_m$
- ☐ C)  $f_{vz} = f_m$
- ☐ D)  $f_{vz} \geq 2f_m$

**Otázka č. 3:** Aký je správny zápis vzťahu pre inverznú DFT?

- ☒ A)  $\sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}nk}$
- ☐ B)  $\frac{1}{N} \sum_{n=0}^{N-1} X(k)e^{j\frac{2\pi}{N}nk}$
- ☐ C)  $\frac{1}{N} \sum_{n=0}^{N-1} X(k)e^{j\omega n}$
- ☐ D)  $\sum_{n=0}^{N-1} x(n)e^{-jkn}$

**Otázka č. 4:** Čo je správne spektrum diskretných signálov je:

- ☒ A) spojité
- ☐ B) diskretné
- ☐ C) frekvenčne neohraničené
- ☐ D) konštatné

A small dialog box in the center says: "Tvoj výsledok je: 25%" with an "OK" button.

At the bottom right, there are two buttons: "Submit" and "Restart".

Obr. 4.7: Finálny návrh testov.

#### 4.4 Implementovanie kódového editora do GUI

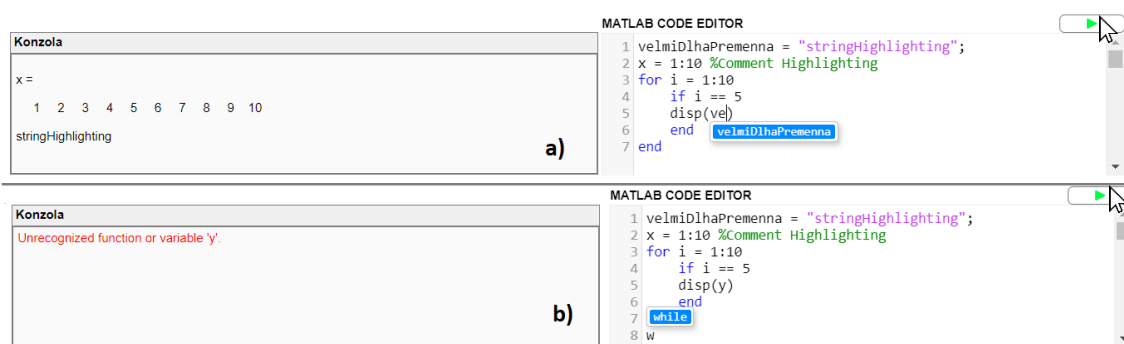
Pred samotným grafickým návrhom časti pre príklady sme museli vytvoriť a navrhnuť vlastný kódový editor, kde môže užívateľ písať MATLAB kód. Vzhľadom k tomu, že MATLAB neposkytuje žiadne textové prvky pre zvýraznenie určitých častí textu, opäť sme sa obrátili na webové jazyky a HTML komponent.

Na zvýraznenie syntaxe jazyka sme využili *open source* JavaScript projekt CodeMirror, ktorý podporuje väčšinu programovacích jazykov. Hoci CodeMirror priamo nepodporuje MATLAB, má ale podporu pre jazyk Octave, ktorý má veľmi podobnú syntax. Preto sme sa rozhodli použiť jazyk Octave a upravili sme ho tak, aby sa zvýrazňovanie syntaxe priblížilo syntaxi jazyka MATLAB.

Okrem zvýraznenia syntaxe sme využili aj ďalšie funkcionality CodeMirror, ktorými boli číslovanie riadkov a automatické dopĺňanie názvov funkcií a premenných. Vzhľadom na odlišné názvy funkcií medzi jazykom Octave a MATLAB sme museli upraviť a prepísať v kóde CodeMirror názvy jednotlivých funkcií. Na automatické dopĺňanie premenných a funkcií sme museli nastaviť klávesovú skratku. Pre túto funkcionality sme zvolili kombináciu kláves *CTRL+SPACE*, pretože táto kombinácia klávesov je použitá nielen v samotnom MATLAB editore, ale aj v mnohých iných kódových editoroch.

Výsledný návrh kódového editora je zobrazený na obr. 4.8 a poskytuje číslovanie riadkov, zvýraznenie syntaxe jazyka a automatické dopisovanie premenných a funkcií.

Na spustenie kódu sme vytvorili tlačidlo s ikonou zeleného trojuholníka, aby bolo podobné tlačidlu, pomocou ktorého sa spúšťa kód v MATLAB editore. Z vytvoreného kódového editora získavame kód vo forme *string* a ten spúšťame prostredníctvom funkcie *evalc*, ktorá okrem spustenia kódu vracia aj výstup z napísaného kódu. Kód napísaný používateľom nemusí byť správny a na vyriešenie tohto problému sme funkciu *evalc* obalili do bloku *try – catch*, kde skúsime spustiť kód a pri nesprávne napísanom kóde zachytávame chybu. Zachytenú chybu a výstup z kódu vypisujeme do konzoly, ktorú sme vytvorili prostredníctvom textového komponentu *Text Area*. Textové výpisy do konzoly sú zobrazené na obr. 4.9 a sú farebne odlíšené, červená symbolizuje výpis chyby a čierna výstup z kódu.

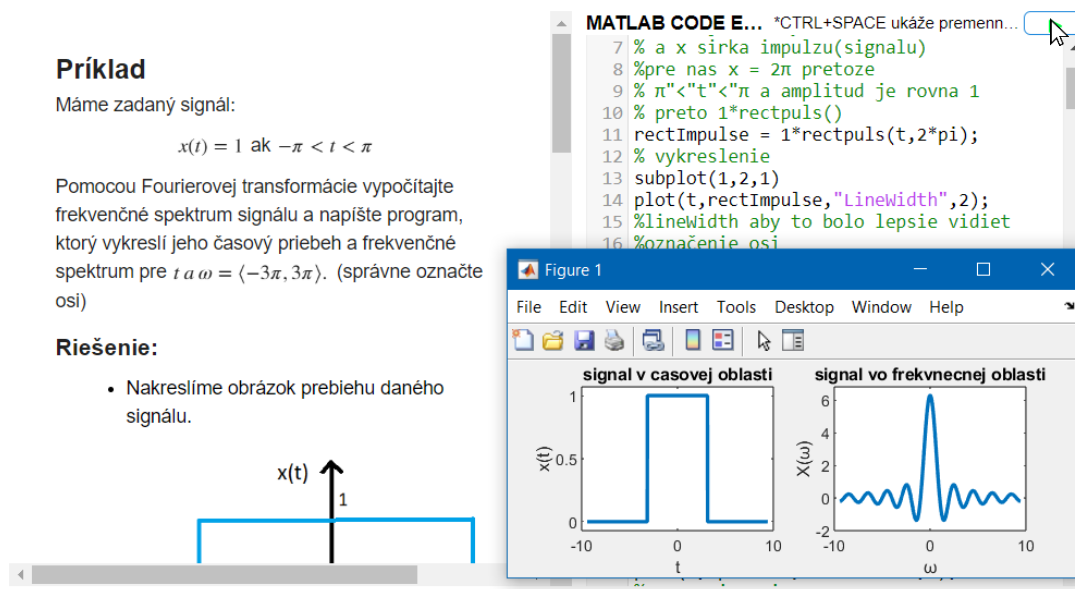


Obr. 4.9: Výpis do konzoly: a) výstup z kódu b) chyba.

## 4.5 Grafický návrh príkladov

Pre návrh príkladov sme si pripravili dva veľmi podobné návrhy. Jeden návrh sme vytvorili pre vyriešené príklady, kde má užívateľ k dispozícii kód a riešenie príkladu. Druhý návrh je určený pre príklady na precvičenie, kde užívateľ môže aktívne pracovať na riešení príkladu a overiť si svoje vedomosti.

Grafický návrh pre vyriešené príklady je zložený z dvoch HTML komponentov, komponentu *Text Area* pre konzolu a komponentu *Button* pre spustenie kódu. Prvý HTML komponent, zobrazuje dokument so zadáním a riešením príkladu, ktorý sme vytvorili prostredníctvom Live Skriptu ako teoretickú časť. Druhý HTML komponent sme vytvorili pre kódový editor, v ktorom je napísaný kód pre vizualizáciu príkladu. Každý riadok nami napísaného kódu sme okomentovali, aby bol lepšie zrozumiteľný pre užívateľa. Po spustení kódu, sa nám výstup z kódu zobrazí v samostatnom okne zobrazenom na obr. 4.9.



Obr. 4.10: Návrh vyriešeného príkladu.

Návrh príkladu na precvičenie je taktiež zložený z dvoch HTML komponentov. Jeden zobrazuje zadanie príkladu bez riešenia, zatiaľ čo druhý zobrazuje kódový editor bez napísaného kódu, čo umožňuje užívateľovi písať vlastný kód. Pre overenie vypočítaného výsledku a napísaného kódu sme vytvorili tlačidlo *Show answer*, ktoré po stlačení ukáže numerický výsledok príkladu a do kódového editoru vypíše kód. Okrem tlačidla *Show answer* sme vytvorili aj tlačidlo *Restart*, ktoré všetko vynuluje.



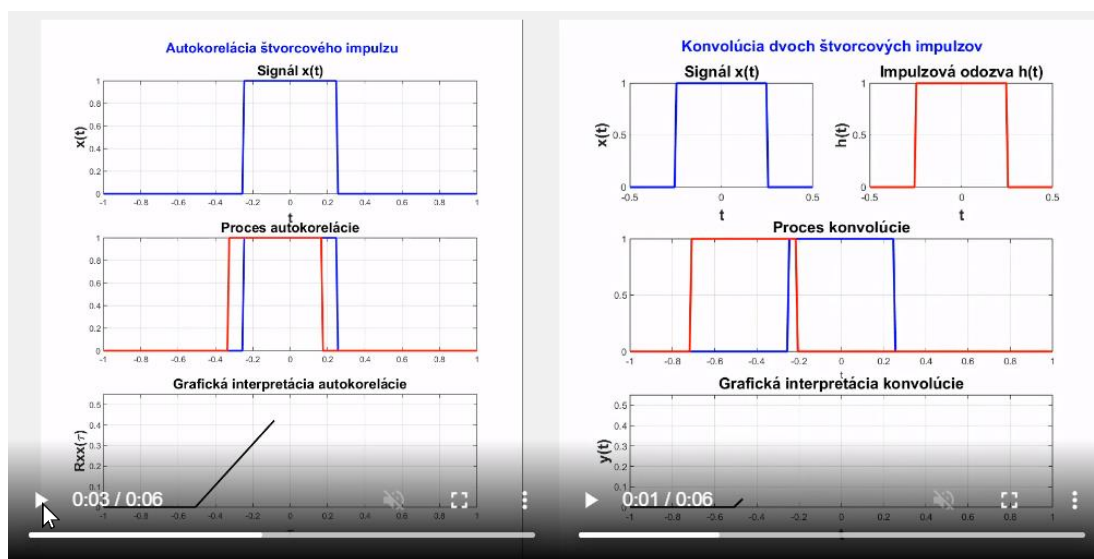
Obr. 4.11: Návrh príkladu na precvičenie.

## 4.6 Tvorba video ukážok a interaktívnych ukážok

Pre vytvorenie video ukážok, sme si v MATLAB vytvorili skripty, ktoré nám graficky vykresľovali meniaci sa priebeh signálov. Pre nahrávanie meniaceho priebehu sme

využili funkciu *VideoWriter*, ktorá zaznamenáva jednotlivé snímky priebehu a vytvorí z toho video súbor vo formáte *MP4*.

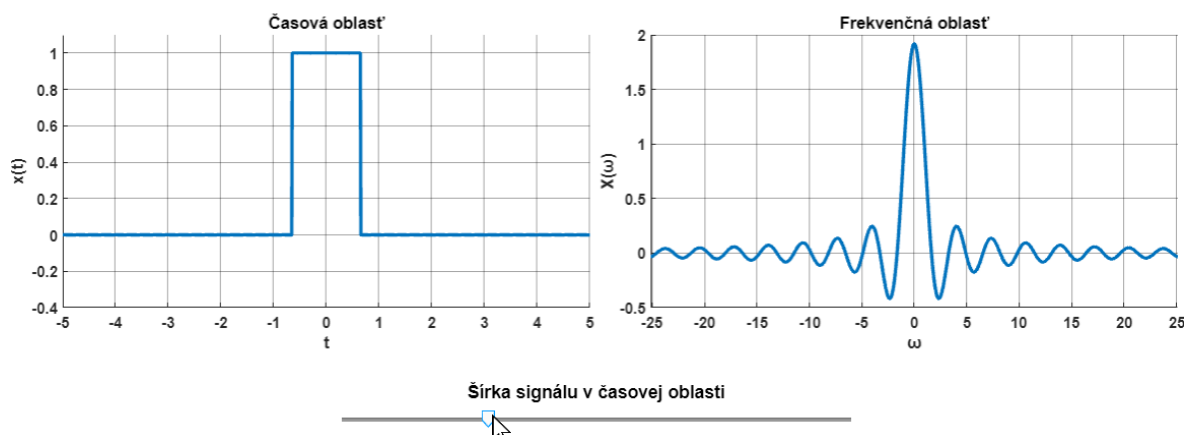
Na zobrazenie vytvoreného videa sme opäť využili HTML komponent, ktorý predstavuje jediný spôsob, ako je možné vložiť video do MATLAB GUI. Je však dôležité poznamenať, že HTML komponent v MATLAB podporuje iba video vo formáte *Ogg*. Avšak súbor formátu *Ogg* nie je možné vytvoriť pomocou funkcie *VideoWriter* a preto sme museli použiť externý softvér VLC media player na konverziu videa z formátu *MP4* na formát *Ogg*. Grafický návrh video ukážky je zobrazený na obr. 4.11.



Obr. 4.12: Návrh video ukážky.

Interaktívna ukážka zobrazená na obr. 4.12 predstavuje grafický priebeh signálu alebo procesu, ktorý sa mení na základe užívateľskej interakcie alebo vstupu od užívateľa. Pre návrh interaktívnych ukážok sme využili komponenty *UIAxes*, do ktorých vykresľujeme priebehy signálov. Pre užívateľskú interakciu sme zvolili hlavné komponenty *Slider*, ale taktiež aj komponenty *Button* alebo *Edit Field*.

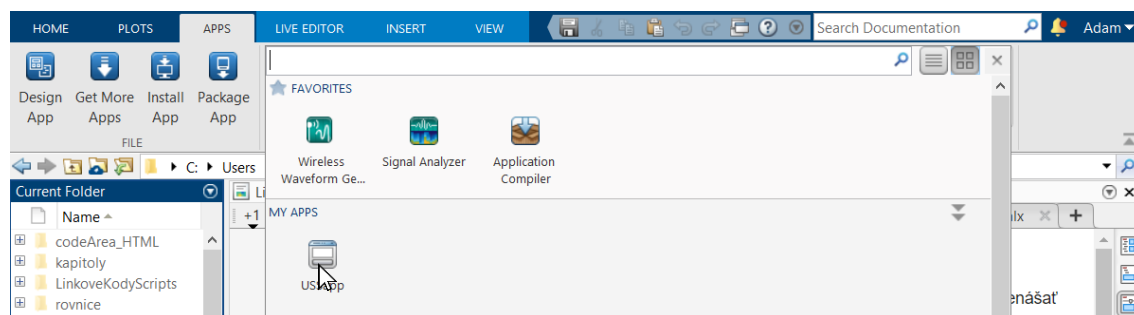
Pre interaktívne ukážky sme museli vytvoriť samostatné funkcie, ktoré sa zavolajú pri kliknutí na tlačidlo alebo pri zmene hodnoty komponentu *Slider* alebo *Edit Field*. Princíp týchto funkcií spočíva v získaní aktuálnej hodnoty z komponentu a na základe tejto hodnoty sú prepočítané dáta podľa špecifikovaného algoritmu vytvoreného pre daný typ ukážky. Pomocou týchto prepočítaných dát je následne aktualizovaný grafický priebeh signálu alebo procesu v komponente *UIAxes*.



Obr. 4.13: Návrh interaktívnej ukážky.

## 4.7 Export vytvorenej aplikácie

Vytvorené GUI sme vyexportovali do dvoch formátov aplikácií. Prvý formát vytvorí MATLAB aplikáciu, ktorá si vyžaduje, aby mal užívateľ nainštalovaný MATLAB, pretože sa nainštaluje priamo do MATLAB záložky *APPS*. Proces inštalácie je jednoduchý a stačí sa riadiť inštalačným návodom. Po inštalácii je aplikáciu možné spustiť kliknutím na ikonu aplikácie v záložke *APPS*.



Obr. 4.14: Nainštalovaná aplikácia v MATLAB APPS.

Pre ďalší formát aplikácie sme museli nainštalovať MATLAB *Compiler*, ktorý nám vytvoril spustiteľný *exe* súbor a ten nám nainštaluje aplikáciu bez potreby MATLAB. Po spustení vyexportovaného súboru sa zobrazí inštalačné menu, kde si autor zvolí miesto na disku kam sa aplikácia a podporné súbory nainštalujú. Pri dodržaní inštalačného návodu sa aplikácia bezproblémovo nainštaluje a po inštalácii je možné aplikáciu spustiť ako akokoľvek inú nainštalovanú aplikáciu.

## 5 APLIKÁCIA SIGNÁLY A SYSTÉMY

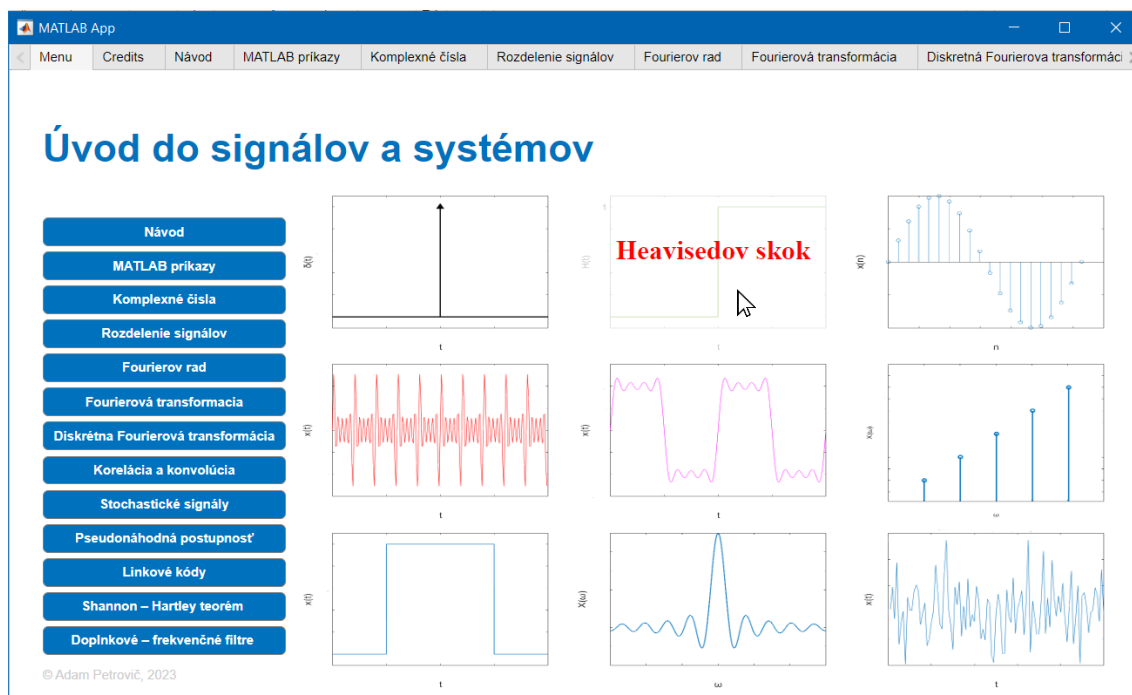
Výsledné GUI je zložené z úvodného menu, kapitoly návod, kapitoly *Credits*, kapitoly základných MATLAB príkazov a 11 kapitolami obsahujúcimi jednotlivé témy. Kapitola s návodom obsahuje stručný popis princípu fungovania testov, kódového editora a návodu pri riešení programovacích príkladov v MATLAB. V kapitole *Credits* sa nachádza meno autora a vedúceho práce, logo univerzity, fakulty a katedry. Jednotlivé témy (s výnimkou doplnkovej témy) korelujú s osnovou predmetu USS. GUI obsahuje 11 kapitol s nasledujúcimi témami:

- Komplexné čísla,
- Rozdelenie signálov,
- Fourierova transformácia,
- Diskrétna Fourierova transformácia a vzorkovanie,
- Konvolúcia a korelácia,
- Stochastické signály,
- Pseudonáhodná postupnosť,
- Linkové kódy,
- Shannon-Hartley teorém,
- Doplnková téma – frekvenčné filtre.

Kapitoly sú ďalej rozdelené na podkapitoly: teoretickú časť, test, interaktívne ukážky, video ukážky a pri kapitolách, kde bolo možné vytvoriť príklady sa nachádzajú aj vyriešené príklady a príklady na precvičenie spojené s vizualizáciou za pomoci jazyka MATLAB.

### 5.1 Úvodné menu

Na obr. 5.1 je zobrazené úvodné menu. Navigovanie medzi jednotlivými kapitolami je zabezpečené prostredníctvom menu tlačidiel s názvami tém alebo prostredníctvom záložkovej lišty, ktorá sa nachádza v hornej časti GUI. V úvodnom menu sa nachádzajú tematické obrázky s ukážkami signálov. Keď kurzorom myši prejdeme cez jeden z týchto obrázkov, zobrazí sa text obsahujúci názov príslušného typu signálu.



Obr. 5.1: Úvodné menu GUI s ukázkami signálov.

## 5.2 Učebné textové materiály

Súčasťou každej kapitoly je podkapitola venujúca sa teoretickému úvodu do danej problematiky. Táto časť pozostáva z textových učebných materiálov, kde sú vysvetlené základné pojmy, princípy, vlastnosti a matematické vzťahy danej témy. Textové materiály sú doplnené obrázkami, pre lepšie pochopenie danej témy.

Textové materiály v teoretickej časti boli napísané s využitím odbornej literatúry, ktorá bola z veľkej časti použitá aj v prvej kapitole tejto práce. Na konci každej textovej časti sa nachádzajú referencie na použitú literatúru.

Na obr. 5.2 je zobrazená ukážka teoretickej časti jednej z kapitol, konkrétne sa jedná o kapitolu DFT a vzorkovanie. Teoretická časť je vždy prvou podkapitolou danej témy, za ktorou zvyčajne nasledujú podkapitoly s interaktívnou alebo video ukážkou, ktoré nadväzujú na túto časť.



## Diskrétna Fourierova transformácia a vzorkovanie

### Vzorkovanie

Vzorkovanie predstavuje jednu z kľúčových operácií pri digitalizácii signálov. Proces vzorkovania spočíva v tom, že zo spojitého signálu  $x(t)$ , pomocou postupnosti impulzov  $p(t)$ , odoberáme vzorky v pravidelných časových intervaloch. Výsledkom tohto procesu je vzorkovaný signál, resp. postupnosť vzoriek  $x(nT_{vz})$  [1].

Matematicky môžeme vzorkovanie vyjadriť:

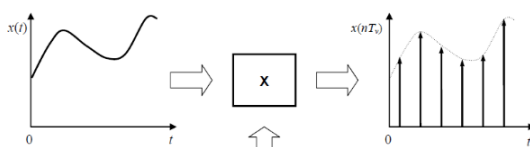
$$x(nT_{vz}) = x(t) \cdot p(t),$$

kde

$t = nT_{vz}$  – časové okamihy vzoriek,

$x(t)$  – spojitý signál,

$p(t)$  – je postupnosť Diracových impulzov.



Obr. 5.2: Teoretická textová časť GUI.

### 5.3 Vytvorené testy

V každej kapitole sa nachádza podkapitola s teoretickým testom. Tieto testy pozostávajú zo 4 otázok. Každá otázka ponúka na výber zo štyroch možností, pričom iba jedna možnosť je správna. Testové otázky sú vytvorené z teoretickej časti a zameriavajú sa primárne na základné pojmy a vzťahy z danej problematiky, ale v kapitolách komplexné čísla a Shannon-Hartley teorém sú niektoré otázky formulované ako príklady. Výsledok testu sa zobrazí v samostatnom okne a je vyjadrený v percentách. Užívateľ môže test vyplniť ľubovoľne veľa krát. Účelom testov je, aby si užívateľ mohol overiť nadobudnuté vedomosti z teoretickej časti a mohol tieto vedomosti následne využiť pri riešení praktických príkladov v ďalších podkapitolách.

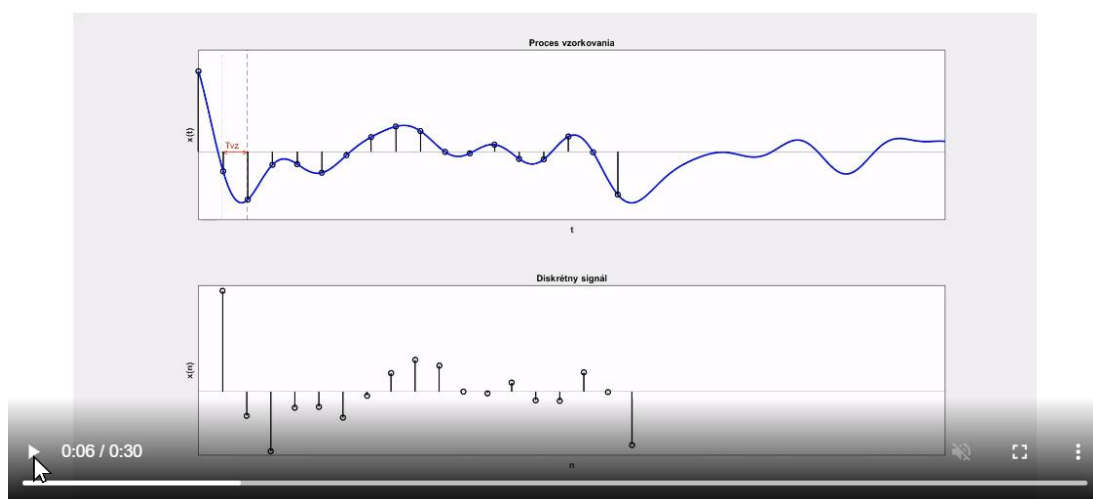
<p>Otázka č.1</p> <p>Aký je správny zápis trigonometrického Fourierovho radu?</p> <p><input checked="" type="checkbox"/> A) <math>x(t) = a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega t) + \sum_{n=1}^{\infty} b_n \sin(n\omega t)</math></p> <p><input type="checkbox"/> B) <math>x(t) = \sum_{n=1}^{\infty} a_n \cos(n\omega t) + \sum_{n=1}^{\infty} b_n \sin(n\omega t)</math></p> <p><input type="checkbox"/> C) <math>x(t) = a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega t) + \sum_{n=1}^{\infty} b_n \sin(n\omega t)</math></p> <p><input type="checkbox"/> D) <math>x(t) = a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega t) - \sum_{n=1}^{\infty} b_n \sin(n\omega t)</math></p>	<p>Otázka č.2</p> <p>Frekvenčné spektrum periodického signálu je:</p> <p><input type="checkbox"/> A) nepárne</p> <p><input type="checkbox"/> B) párne</p> <p><input type="checkbox"/> C) párne a nekonečne široké</p> <p><input type="checkbox"/> D) neharmonické</p>
<p>Otázka č.3</p> <p>Správny zápis vzorca pre výpočet jednosmernej zložky je:</p> <p><input type="checkbox"/> A) <math>\frac{2}{T} \int_0^T x(t) \cos(n\omega t) dt</math></p> <p><input checked="" type="checkbox"/> B) <math>\frac{2}{T} \int_0^T x(t) \cos(n\omega t) dt</math></p> <p><input type="checkbox"/> C) <math>\frac{1}{T} \int_0^T x(t) \sin(n\omega t) dt</math></p> <p><input type="checkbox"/> D) <math>\frac{1}{T} \int_0^T x(t) dt</math></p>	<p>Otázka č.4</p> <p>Fourierov rad pre nepárne periodické funkcie obsahuje len:</p> <p><input type="checkbox"/> A) sínusové členy</p> <p><input checked="" type="checkbox"/> B) jednosmernú zložku</p> <p><input type="checkbox"/> C) sínusové členy a kosínusové členy</p> <p><input type="checkbox"/> D) kosínusové členy</p>

Obr. 5.3: Testy v GUI.

## 5.4 Video ukážky signálov a procesov

Video ukážky sú súčasťou len kapitol venujúcim sa DFT a vzorkovanie, konvolúcii, korelácii. V kapitole DFT a vzorkovanie je video ukážka zameraná na proces vzorkovania, kde video zobrazuje ako zo spojitého signálu vzniká procesom vzorkovania diskretný signál a následne pokračuje vizualizáciou výpočtu frekvenčného spektra s využitím matematickej operácie DFT.

V kapitole konvolúcia a korelácia je k dispozícii šesť video ukážok, ktoré zobrazujú grafickú interpretáciu procesov konvolúcie, autokorelácie a krížovej korelácie. Video ukážka vzorkovania je zobrazená na obr. 5.4 a video ukážka z kapitoly konvolúcie a korelácie je zobrazená na obr. 4.11 v predošlej kapitole.

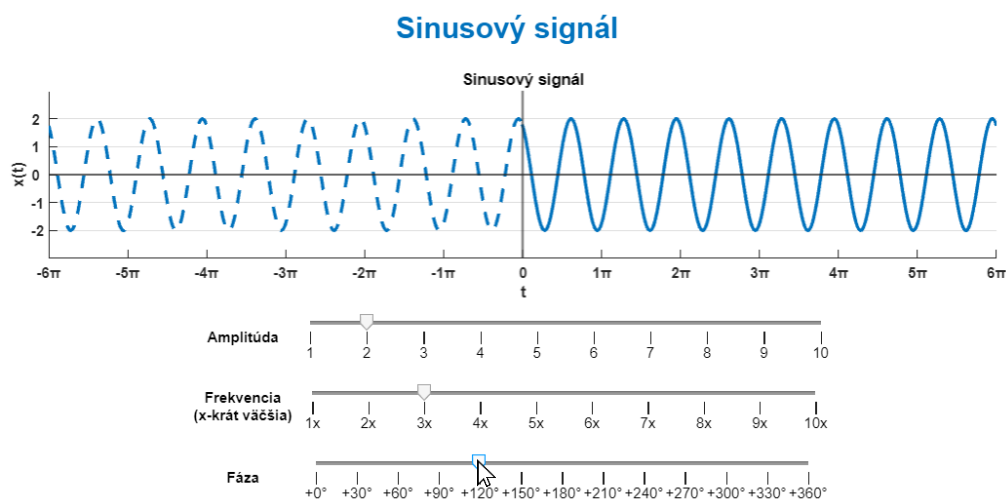


Obr. 5.5: Video ukážka – proces vzorkovania.

## 5.5 Interaktívne ukážky signálov a systémov

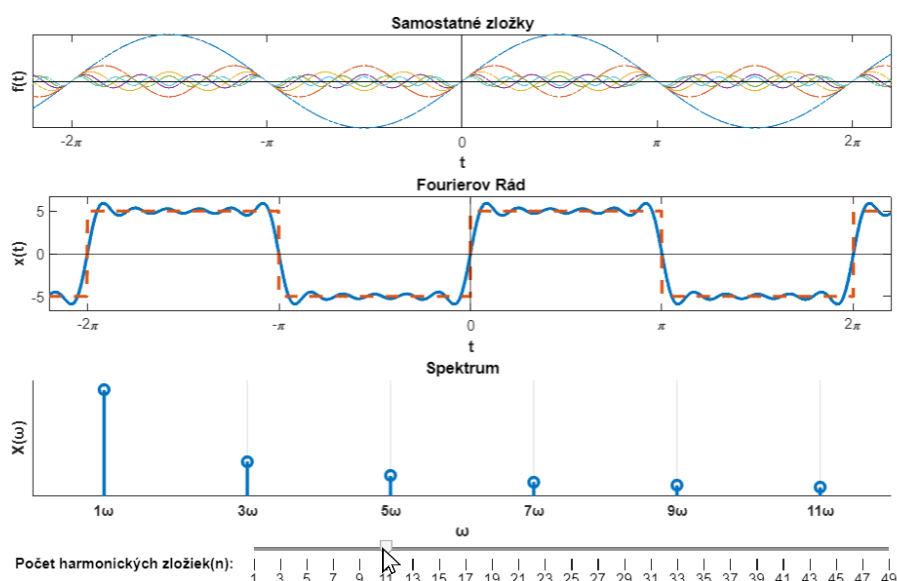
Interaktívne ukážky sú s výnimkou kapitol venujúcim sa komplexným číslam, korelácii, konvolúcii, Shannon-Hartley teorému a doplnkovej témy - frekvenčné filtre súčasťou každej kapitoly.

Prvá kapitola, ktorá obsahuje interaktívnu ukážku je kapitola rozdelenie signálov. V tejto ukážke je možné pomocou troch komponentov *Slider* meniť amplitúdu, fázu a frekvenciu sínusového signálu. Interaktívna ukážka sínusového signálu je zobrazená na obr. 5.5 a slúži na demonštráciu základných parametrov signálu v časovej oblasti.



Obr. 5.5: Interaktívna ukážka sínusového signálu.

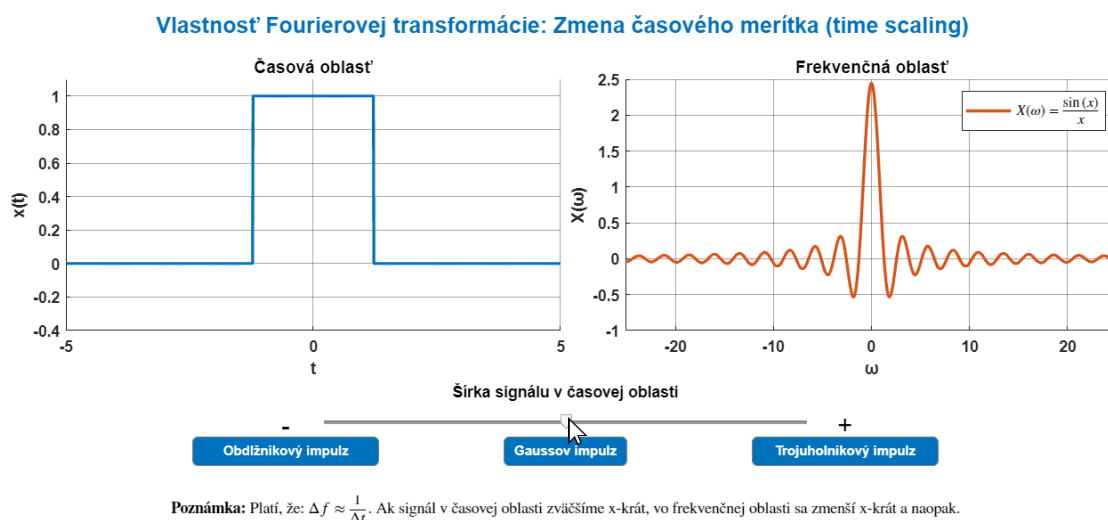
Na obr. 5.6 je zobrazená interaktívna ukážka kapitoly FR, ktorá vizuálne zobrazuje vznik obdĺžnikového signálu. V tejto ukážke je možné pomocou komponentu *Slider* pridávať a odoberať harmonické zložky obdĺžnikového signálu. Na základe zmeny hodnoty komponentu *Slider* sa mení priebeh signálu v časovej aj vo frekvenčnej oblasti. Okrem výsledného signálu sú vizuálne zobrazené aj samostatné harmonické zložky, z ktorých je vytvorený výsledný obdĺžnikový signál.



Obr. 5.6 Interaktívna ukážka FR.

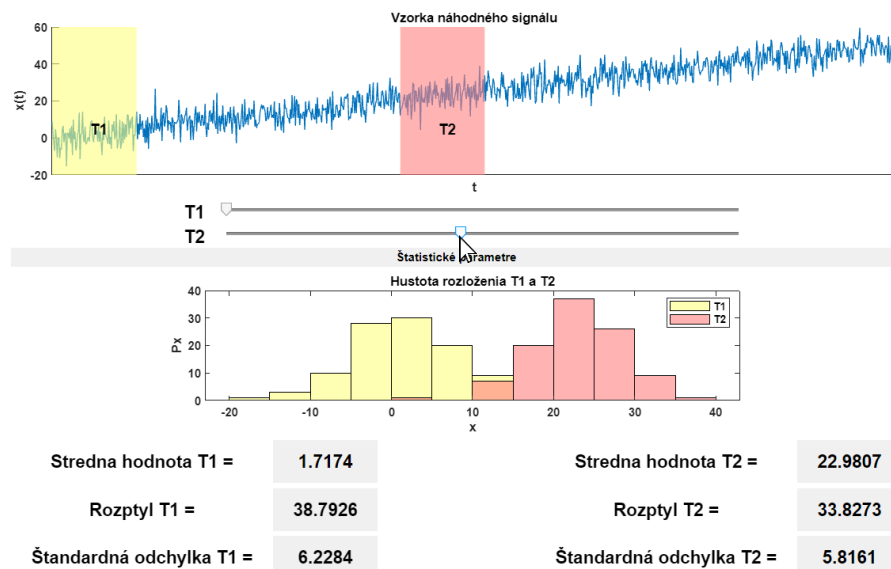
Interaktívna ukážka v kapitole FT zobrazuje jednu z vlastností FT – zmenu časového merítka. Užívateľ má možnosť meniť šírku signálu v časovej oblasti a sledovať, ako sa bude meniť šírka signálu vo frekvenčnej oblasti. Užívateľ môže sledovať zmenu na troch typoch signálov (obdĺžnikový, trojuholníkový a Gaussov impulz), pričom zmenu typu signálu je možné vykonávať prostredníctvom tlačidiel.

Frekvenčné spektrum daného typu signálu je opísané aj matematickým vzťahom a je zobrazené v pravej hornej časti grafického komponentu *UIAxes*. Pod jednotlivými priebehmi sa nachádza poznámka, ktorá matematicky opisuje zmenu šírky v spektrálnej oblasti pri zmene šírky signálu v časovej oblasti, čím sa uľahčuje pochopenie interakcie medzi časovou a frekvenčnou doménou signálu.



Obr. 5.7: Interaktívna ukážka vlastnosti FT – zmena časového merítka.

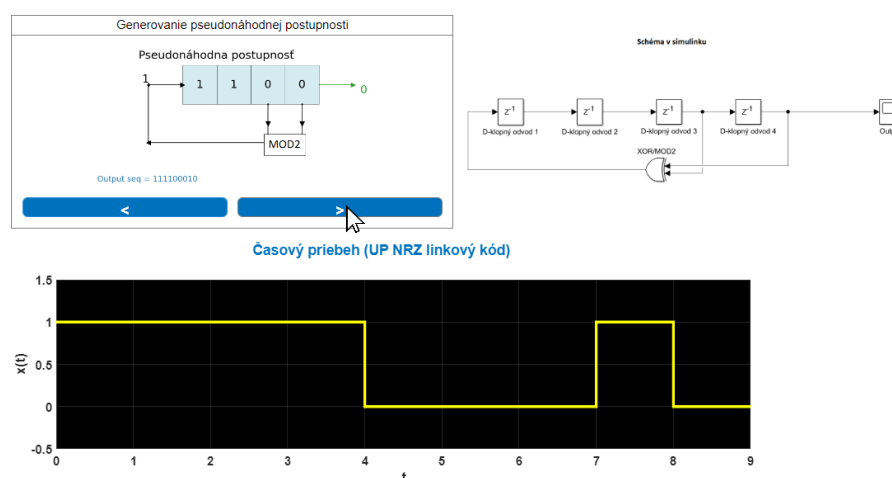
Kapitola o stochastických signáloch obsahuje dve ukážky. Obe ukážky majú rovnakú grafickú podobu, pričom jedna zobrazuje priebeh nestacionárneho stochastického signálu a druhá priebeh stacionárneho stochastického signálu. Na obr. 5.8 je zobrazená ukážka nestacionárneho signálu, pomocou komponentov *Slider* je možné pohybovať časovými oknami  $T1$ ,  $T2$  a sledovať ako sa menia štatistické parametre signálu. Hustota rozloženia pravdepodobnosti je vyjadrená histogramom a ostatné parametre sú vyjadrené numericky v samostatných textových poliach. Účelom týchto ukážok je demonštrovať rozdiel medzi nestacionárnym a stacionárnym stochastickým signálom. Pri pohybe časovými oknami v ukážke nestacionárneho signálu sa štatistické hodnoty menia, zatiaľ čo pri pohybe oknami v ukážke stacionárneho signálu sa štatistické hodnoty nemenia.



Obr. 5.8: Interaktívna ukážka nestacionárneho stochastického signálu.

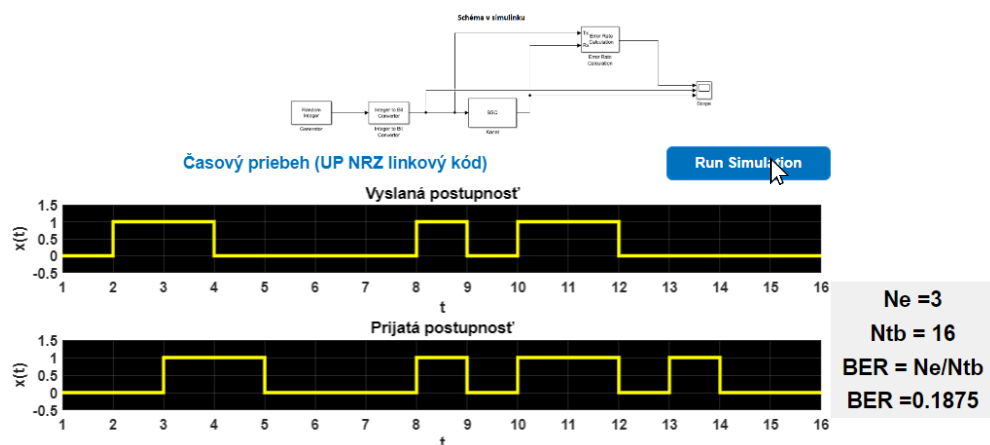
V kapitole zaoberajúcej sa pseudonáhodnou postupnosťou (PP) sa nachádzajú dve ukážky, ktoré boli pôvodne vytvorené v MATLAB SIMULINK. Avšak, kvôli absencii SIMULINK *Compiler* v MATLAB nebolo možné GUI exportovať ako samostatnú aplikáciu. Preto sme simulácie zo SIMULINK preprogramovali čisto pomocou MATLAB kódu. Napriek tomu, pre demonštráciu princípov, sú v tejto časti zahrnuté obrázky so zapojenými schémami zo SIMULINK.

Prvá ukážka, simuluje generovanie PP, pomocou tlačidiel môžeme pridávať alebo odoberať bity. Princíp činnosti generátora PP je zobrazený prostredníctvom obrázkovej schémy zobrazenej na obr. 5.9. Táto obrázková schéma sa mení pri pridaní alebo odstránení bitu. Okrem obrázkovej schémy sa mení vizualizácia časového priebehu PP, ktorá sa nachádza pod obrázkom schémy zo SIMULINK a schémou generátora PP.



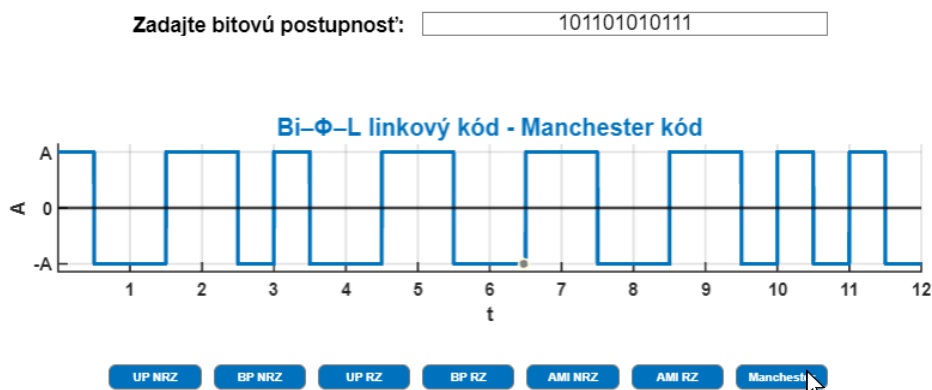
Obr. 5.9: Interaktívna ukážka generovania PP.

Druhá ukážka simuluje chybovosť komunikačného kanála, cez ktorý prenášame bitovú postupnosť vygenerovanú PP. Po stlačení tlačidla *Run simulation* sa zobrazia dva časové priebehy, kde prvý zobrazuje bitovú postupnosť pred vyslaním cez komunikačný kanál a druhý zobrazuje prijatú bitovú postupnosť. Počet zle prijatých a celkovo vyslaných bitov je vypísaný v textovom poli. Z týchto hodnôt je následne vypočítaná celková chybovosť (BER – Bit Error Rate). Počet a časová pozícia chybne prijatých bitov, je náhodne vygenerovaná po každom kliknutí na tlačidlo *Run simulation*. Ukážka je zobrazená na obr. 5.10 a jej cieľom je oboznámiť užívateľa s využitím PP pri meraní a výpočte chybovosti komunikačného kanála.



Obr. 5.10: Interaktívna ukážka merania chybovosti kanála.

Poslednou kapitolou, ktorá obsahuje interaktívnu ukážku je kapitola linkové kódy. Táto ukážka umožňuje užívateľovi zadať bitovú postupnosť, ktorá je následne transformovaná na vybraný typ linkového kódu. Typ linkového kódu je možné meniť prostredníctvom tlačidiel s názvom daného linkového kódu. Pri zmene linkového kódu sa zadaná bitová postupnosť nevymaže, čo umožňuje užívateľovi sledovať a vizuálne porovnať rozdiely medzi jednotlivými typmi linkových kódov.



Obr. 5.11: Interaktívna ukážka linkových kódov.

## 5.6 Signály a systémy v príkladoch

Príklady sú okrem kapitol PP a linkové kódy súčasťou každej kapitoly. Tieto príklady sú rozdelené do dvoch kategórií. Prvá kategória zahŕňa čisto programovacie príklady pre vizualizáciu signálov a vykreslenie grafov závislostí. Druhá kategória príkladov obsahuje numerický príklad, kde okrem samotného výpočtu je úlohou pomocou MATLAB kódu vizualizovať priebeh daného signálu.

Pri každej téme je ako prvým príkladom, vždy ukážkový vyriešený príklad. V prípade iba čisto programovacích príkladov je zobrazené len zadanie a priamo kód, ktorý je možné hneď spustiť. Numerické príklady okrem kódu obsahujú aj postup riešenia. V oboch prípadoch je kód doplnený komentármi, ktoré zjednodušujú jeho pochopenie. Ukážka vyriešeného príkladu a kódu z témy DFT a vzorkovanie je zobrazená na obr. 5.12.

**Príklad**

Vypočítajte spektrum diskrétného signálu  $x(n) = [3, 1, 2, -1]$ . Napíšte program, ktorý overí výpočet a graficky vykreslí časovú oblasť signálu  $x(n)$  a frekvenčné spektrum signálu  $X(k)$ . (označte osi)

**Riešenie:**

- Ako prvé potrebujeme zistiť počet vzoriek  $N$ . V tomto prípade je  $N = 4$ . Následne môžeme dosadiť do vzorca pre DFT.

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}nk} = \sum_{n=0}^3 x(n)e^{-j\frac{2\pi}{4}nk} = \sum_{n=0}^3 x(n)e^{-j\frac{\pi}{2}nk}$$

- Následne budeme vypočítavať jednotlivé koeficienty DFT pre  $k = 0, 1, \dots, N-1$  a začínať budeme v nule ( $X(0)$ ) a potom budeme pokračovať až po  $X(N-1)$ .

Pre  $k = 0$ :

$$X(0) = \sum_{n=0}^3 x(n)e^{-j\frac{\pi}{2}n \cdot 0} =$$

**MATLAB CODE EDITOR** \*CTRL+SPACE ukáže premenné a funkcie

```
1 %vytvoríme pole pre signal v
2 %casovej oblasti zo zadania
3 xn = [3,1,2,-1];
4 %rozdelenie plotu na 2 casti vyberame 1 okno
5 subplot(1,2,1)
6 %vykreslime
7 stem(xn)
8 %oznacenie osi x
9 xlabel("n")
10 %oznacenie osi y
11 ylabel("x(n)")
12 %nastavenie nazvu plotu
13 title("signal v casovej oblasti")
14
15 %spektrum
16 %vyuzijeme funkciu fft()
17 %ktora je zrychleny algoritmus DFT
```

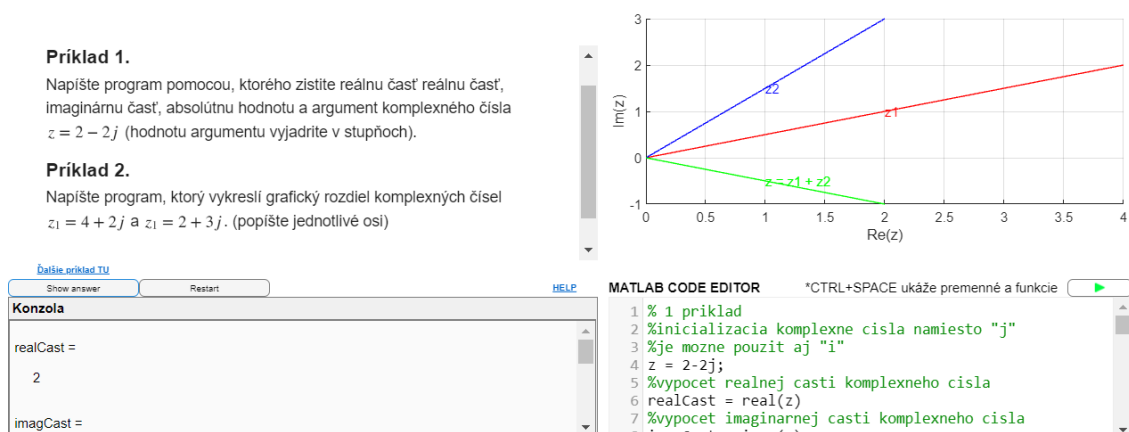
**Konzola**

```
Xk =
5.0000 + 0.0000i 1.0000 - 2.0000i 5.0000 + 0.0000i 1.0000 + 2.0000i
```

Obr. 5.12: Vyriešený príklad z kapitoly DFT.

Po ukážkových riešených príkladoch, sú k dispozícii príklady na precvičenie, ktoré majú podobné zadanie a kde úlohou užívateľa je, podľa zadania, vyriešiť daný príklad. Pre kontrolu vypočítaného výsledku a vykresleného priebehu, je vytvorené tlačidlo *Show answer*, ktoré okrem zobrazenia numerického výsledku do textového poľa, vypíše a spustí predpripravený kód.

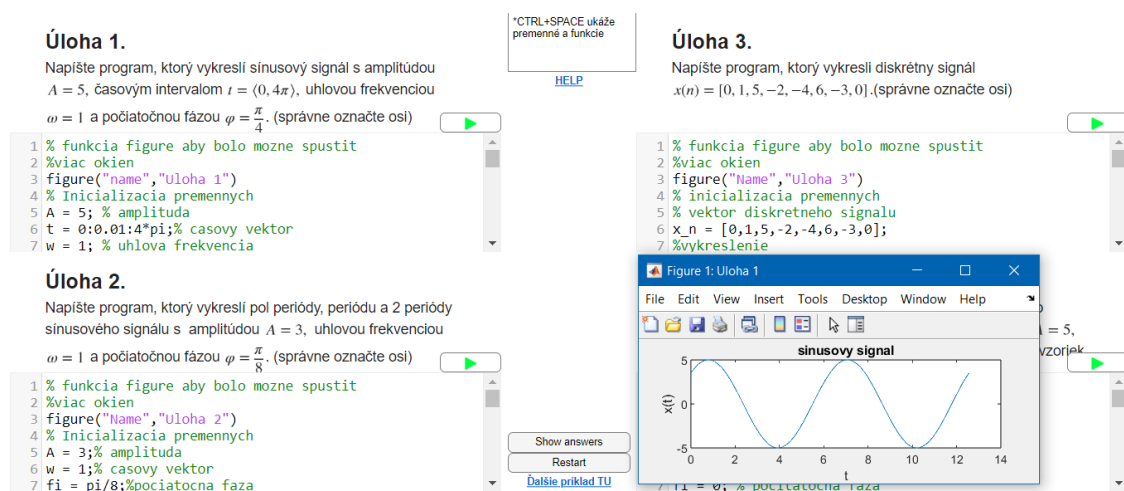
Na obr. 5.13 sú zobrazené programovacie príklady na precvičenie z témy komplexné čísla. V prvom príklade je zadané komplexné číslo a úlohou užívateľa je napísať program, ktorý vypíše do konzoly reálnu časť, imaginárnu časť, absolútnu hodnotu a argument zadaného komplexného čísla. V druhom príklade je úlohou napísať program, ktorý vykreslí graficky rozdiel komplexných čísel zo zadania.



Obr. 5.13: Programovací príklad na komplexné čísla.

Tlačidlo *HELP* zobrazené na obr. 5.13 slúži v prípade, že užívateľ potrebuje pomôcť s nejakou MATLAB funkciou alebo nevie, ako postupovať pri riešení príkladov. Tlačidlo *HELP* sa nachádza pri každom príklade na precvičenie a po jeho stlačení sa zobrazí okno, ktoré obsahuje kapitolu s návodom a kapitolu s tabuľkou MATLAB príkazov.

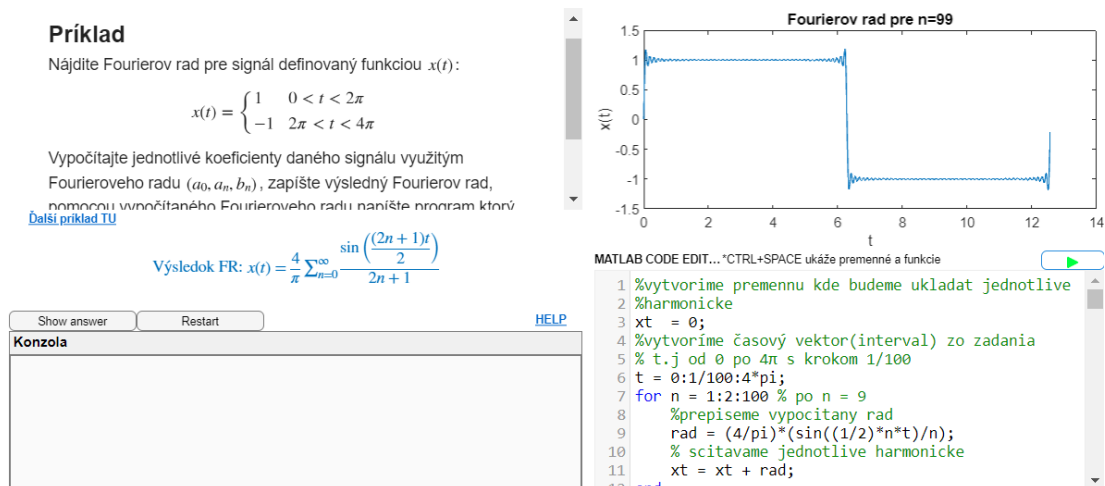
Pri téme, ktorá sa zaoberá rozdelením signálov je k dispozícii viacero príkladov, ktoré patria do kategórie programovacích príkladov. Úlohou v týchto príkladoch je na základe zadaných parametrov vykresliť priebehy signálov.



Obr. 5.14: Programovacie príklady na vizualizáciu signálov.

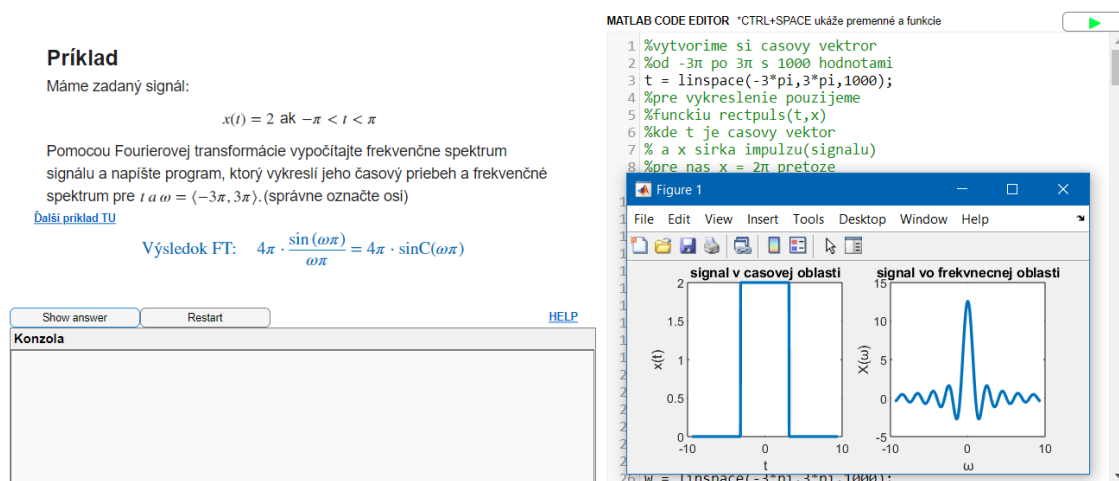
V kapitole FR je v príkladoch úlohou najprv numericky vypočítať FR a potom na základe vypočítaného výsledku napísať program pre vizualizáciu signálu v časovej oblasti. Na obrázku 5.15 je zobrazený príklad FR spolu s výsledkom, ktorý získame po stlačení tlačidla *Show Answer*.





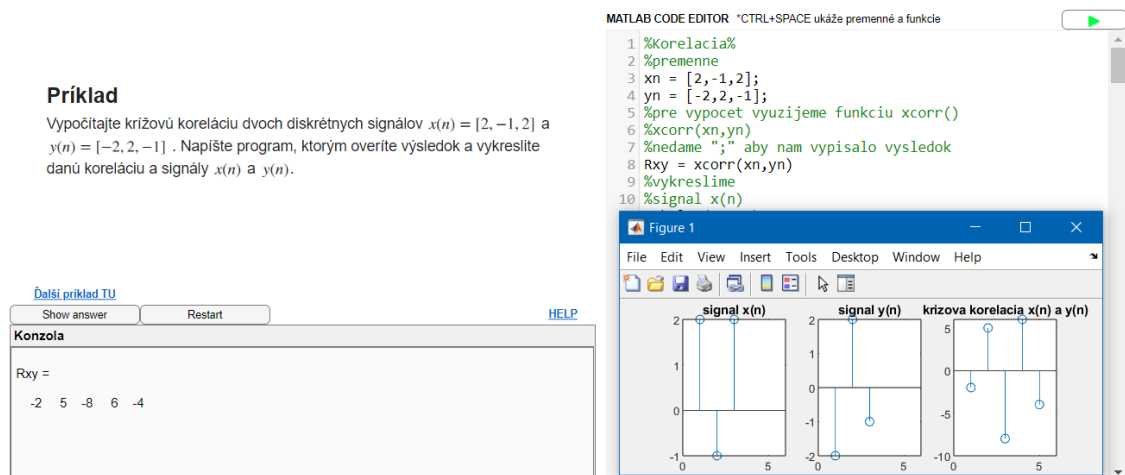
Obr. 5.15: Príklad na FR s vizualizáciou.

V príkladoch pri témach FT a DFT je úlohou zo zadaného signálu v časovej oblasti, vypočítať jeho frekvenčné spektrum. Následne vytvoriť program, ktorý vykreslí signál v časovej aj frekvenčnej oblasti. Zadanie príkladu z FT aj s riešením je zobrazené na obr. 5.16.



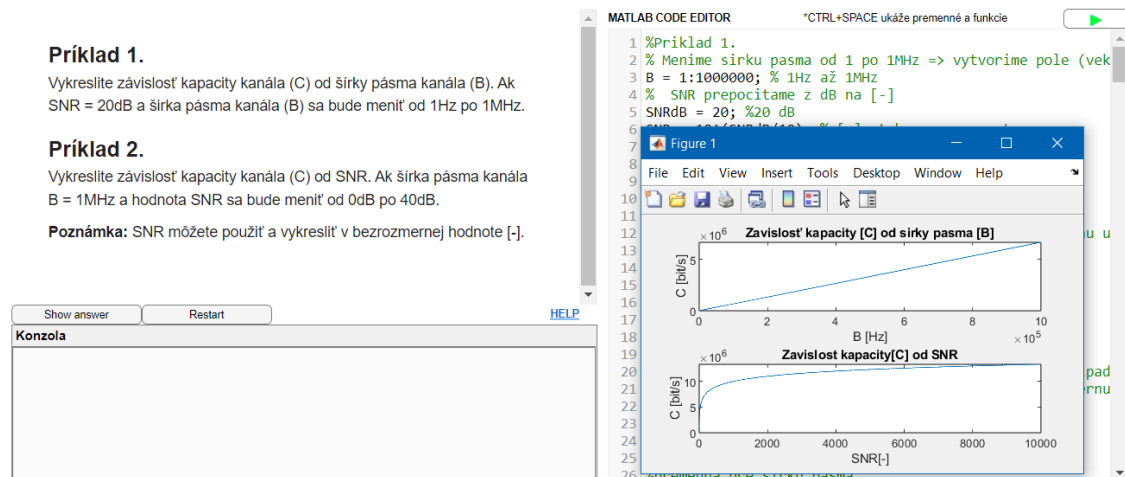
Obr. 5.16: Príklad na FT s vizualizáciou.

V kapitole venujúcej sa konvolúcii a korelácii je vytvorený osobitný príklad pre konvolúciu a osobitný pre koreláciu. V zadaní týchto príkladov je požadované vypočítať diskretnú konvolúciu alebo koreláciu dvoch signálov, napísať program, ktorý overí numerický výpočet a vykreslí oba signály a výsledok konvolúcie alebo korelácie. V týchto príkladoch nie je potrebné po stlačení tlačidla *Show answer*, zobrazovať výsledok v samostatnom textovom poli, pretože výsledok bude zobrazený prostredníctvom kódu v konzole.



Obr. 5.17: Príklad na krížovú koreláciu s vizualizáciou.

Pre kapitolu Shannon – Hartley teorém, sú k dispozícii dva príklady zobrazené na obr. 5.18. Prvý príklad je zameraný na vykreslenie grafu závislosti kapacity kanála od šírky pásma. Druhý príklad je podobného charakteru a to vykreslenie závislosti kapacity kanála od parametra odstup signálu od šumu (SNR – Signal to Noise Ratio). Účelom týchto príkladov je uvedomiť si, ako sa zvyšuje kapacita kanála v závislosti od šírky pásma a SNR.



Obr. 5.18: Príklad na kapacitu kanála.

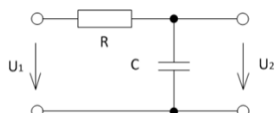
Pri doplnkovej téme – frekvenčné filtre sú k dispozícii dva príklady pre každý typ analógového filtra (filter typu dolný priepust, horný priepust, pásmový priepust a pásmový zadrž). V príkladoch je úlohou podľa zadanej obvodovej schémy pomocou MATLAB kódu znázorniť amplitúdovú a fázovú charakteristiku filtra, zistiť a vyznačiť na grafe hodnotu medznej frekvencie, meniť hodnoty obvodových prvkov a sledovať zmenu medznej frekvencie. Ukážkový príklad je zobrazený na obr. 5.19. Cieľom tejto

doplnkovej kapitoly je rozšíriť vedomosti užívateľa o problematiku analógových filtrov, pretože táto problematika sa u nás na katedre už nevyučuje.

#### Zadania úloh pre filtre typu dolný priestup

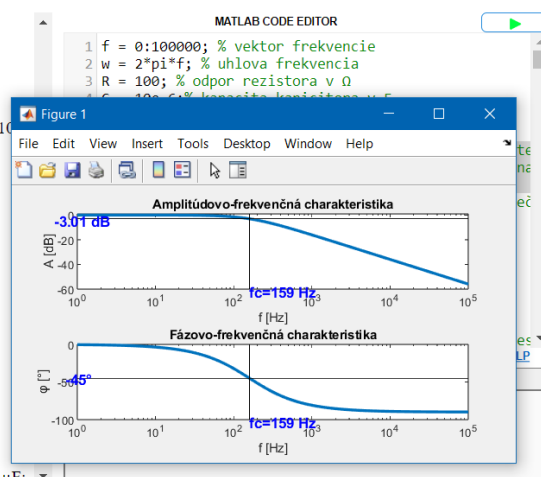
##### 1. Úloha

Zo zadanej obvodovej schémy RC DP s hodnotami prvkov:  $R = 100\Omega$  a  $C = 10\mu F$  riešte úlohy v matematickom prostredí MATLAB:



Obr.3.4 Obvodová schéma RC DP

- znázorníte amplitúdovú a fázovú charakteristiku; použijete funkciu `semilogx()`,
- zistíte hodnotu medznej frekvencie a vyznačíte ju na grafe z úlohy a),
- ako sa zmení medzná frekvencia, ak:
  - hodnoty rezistora za budú meniť v rozsahu od  $1\Omega$  po  $1k\Omega$  a hodnota kapacitora sa nezmení;
  - hodnoty kapacitora meníte v rozsahu  $100pF$  po  $100\mu F$ ;



Obr. 5.19: Príklady na frekvenčné filtre.

Pre užívateľa sme takmer v každej kapitole pripravili aj ďalšie doplnkové príklady, ktorých zadania sa zobrazia v samotných oknách po stlačení tlačidla *d'alsie príklady TU*. Tieto doplnkové príklady, ak nepatria do kategórie čisto programovacích príkladov, obsahujú aj riešenie a sú z matematickej stránky o niečo zložitejšie.

**Príklad**

Nájdite Fourierov rad pre signál definovaný

$$x(t) = \begin{cases} 1 & 0 < t \\ -1 & 2\pi < \end{cases}$$

Vypočítajte jednotlivé koeficienty daného s radu  $(a_0, a_n, b_n)$ , zapíšte výsledný Fourier

[Ďalší príklad TU](#)

Show answer Restart

Konzola

**Doplnkový príklad**

**Príklad**

Nájdite Fourierov rad pre signál definovaný funkciou  $f(t)$ :

$$f(t) = t, \quad -\pi < t < \pi$$

Vypočítajte jednotlivé koeficienty daného signálu využitím Fourieroveho radu  $(a_0, a_n, b_n)$ , zapíšte výsledný Fourierov rad a pomocou vypočítaného Fourieroveho radu napíšte program, ktorý graficky vykreslí zadaný signál. (popíšte jednotlivé osi)

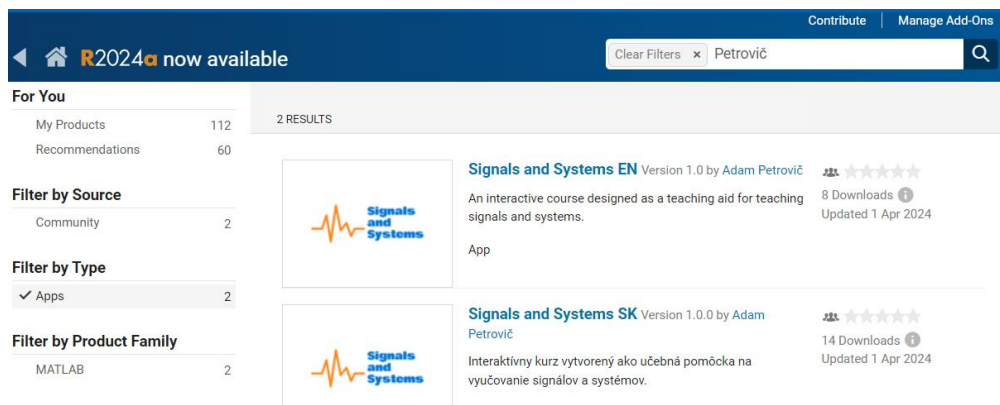
**Riešenie:**

Obr. 5.20: Doplnkové príklady.

## 5.7 Technické problémy a testovanie

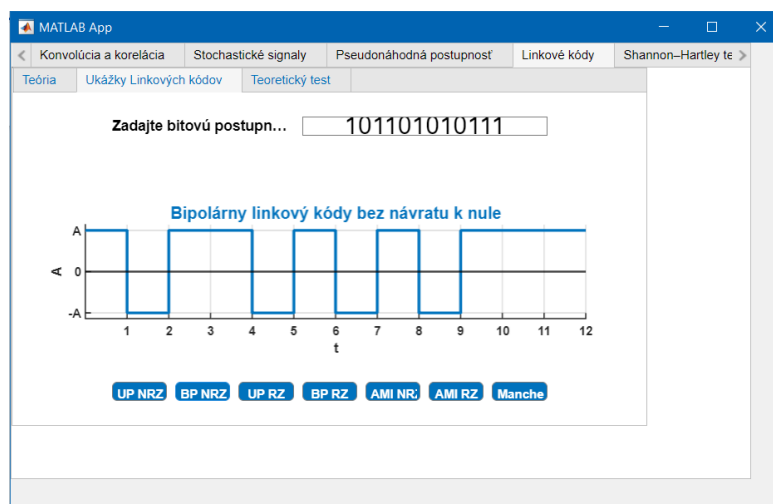
Verzia aplikácie, ktorá sa nainštaluje priamo do MALTAB APPS, je multiplatformová a spustiteľná na akomkoľvek operačnom systéme, ktorý podporuje verziu MALTAB 2023a a vyššiu. Avšak aplikácia vo formáte *exe* je spustiteľná len na operačnom systéme Windows. Aplikáciu sme preložili aj do anglického jazyku aby bola prístupná aj výmenným študentom zo zahraničia. Anglická verzia aplikácie je obsahovo totožná s

jediným rozdielom, že teoretická textová časť nie je vytvorená v HTML, ale vo formáte PDF. Na zjednodušenie prístupu a zdieľania sme aplikáciu nahrali na GitHub a MATLAB File Exchange. Verziu aplikácie, ktorá sa inštaluje do MATLAB, je možné nainštalovať priamo z MATLAB *Add-On Explorer*, odkiaľ sa inštalujú aj iné *Toolbox*.



Obr. 5.21: Nahratá aplikácia na MATLAB File Exchange.

Najväčšou nevýhodou GUI je čas spustenia. Po spustení aplikácie musíme čakať určitý čas, kým sa GUI načíta, čo môže užívateľovi indikovať, že aplikácia zamrzla. Podobný problém sa objavuje aj pri zväčšovaní alebo zmenšovaní okna GUI zobrazenom na obr. 5.21. Okno GUI sa zväčšuje rýchlejšie ako jeho grafické komponenty, čo na určitý časový okamih spôsobuje, že GUI sa zdá neresponzívne. Tieto problémy sme neboli schopní odstrániť a sú spôsobené tým, že jazyk MATLAB je vo všeobecnosti pomalý a nie je navrhnutý pre veľké aplikácie s veľkým počtom grafických komponentov.



Obr. 5.22: Technický problém GUI.

Ďalšou nevýhodou, ktorá vznikla zvolením jazyka MATLAB, je potreba relatívne veľkého množstva systémových zdrojov. Na svoju relatívne malú veľkosť si vyžaduje

pomerne veľké množstvo pamäte RAM, čo môže byť pre niektoré systémy a používateľov obmedzujúce. Tento fakt môže ovplyvniť výkon a plynulosť používania GUI, najmä na zariadeniach s obmedzenými systémovými zdrojmi.

Testovacia verzia aplikácie, ktorá zahŕňala prvých päť kapitol, bola zverejnená pre študentov predmetu Úvod do signálov a systémov ako učebná pomôcka pred skúškovým obdobím. Na získanie spätnej väzby o aplikácii sme vytvorili online formulár pomocou Google Forms. Bohužiaľ, formulár vyplnili iba dvaja študenti, čo nám neumožnilo vytvoriť záver. Avšak spätnú väzbu sme získali aj prostredníctvom správy na Microsoft Teams, kde nám študent z fakulty riadenia a informatiky uviedol, že aplikáciu vo formáte *exe* nie je možné spustiť kvôli chybnéj ceste k súboru v kóde. Táto chyba je pravdepodobne dôvodom, prečo formulár nevyplnilo viacero študentov. Vo finálnej verzii aplikácie sme túto chybu opravili a prekontrolovali sme cesty ku všetkým súborom.

## **5.8 Porovnanie vytvoreného GUI s bezplatne dostupnými kurzami**

Bezplatné kurzy, s výnimkou mobilných aplikácií, sú k dispozícii ako webové stránky alebo webové aplikácie, čo znamená, že ich je možné používať na akomkoľvek operačnom systéme a zariadení s prístupom k internetu a bez nutnosti inštalácie. Nami vytvorená aplikácia si však vyžaduje inštaláciu a verzia, ktorá sa nainštaluje do MATLAB *APPS*, je síce multiplatformová ale je dostupná len na operačné systémy, ktoré majú podporu pre plnohodnotnú verziu MATLAB. V prípade, že užívateľ nemá prístup k MATLAB alebo nemá nainštalovaný MATLAB je k dispozícii druhá verzia aplikácie, ktorej dostupnosť je však obmedzená iba na operačný systém Windows.

Kurzy od MIT, ČVUT a Wolfram zahŕňajú aj video prednášky, kde odborní pedagógovia detailne vysvetľujú jednotlivé témy. Naša aplikácia však túto možnosť neposkytuje, pretože bola navrhnutá len ako doplnková výučbová pomôcka a jej cieľom nie je nahrádzať prednášky. Napriek tomu poskytuje množstvo interaktívnych ukážok a niekoľko video ukážok, ktoré okrem kurzu od Wolfram neposkytoval žiadny iný kurz, čo prispieva k lepšiemu a hlbšiemu pochopeniu danej problematiky.

Okrem kurzov MIT a Wolfram, žiadne iné dostupné kurzy neobsahovali príklady. Kurz od MIT sa zameriaval iba na výpočtové príklady, zatiaľ čo kurz od Wolfram poskytoval príklady aj s vizualizáciou, ale využíval jazyk Wolfram, ktorý nie je medzi študentmi veľmi rozšírený. Nami vytvorená aplikácia preto obsahuje kombináciu výpočtových príkladov aj s vizualizáciou signálov za pomoci jazyka MATLAB, ktorý je

medzi študentmi viacej rozšírený, pretože je vyučovaný ako jeden z povinných predmetov a okrem toho sa využíva aj na samotných cvičeniach predmetu Úvod do signálov a systémov. Taktiež výpočtové príklady a programovacie príklady obsiahnuté v aplikácii boli inšpirované a vytvorené na podobný princíp ako príklady, ktoré sú preberané na cvičeniach predmetu Úvod do signálov a systémov. Vytvorené príklady sú vhodné ako materiál pre cvičenia, alebo ako doplnkové príklady na ďalšie precvičovanie a rozvíjanie vedomostí.

V našej aplikácii sú zahrnuté testy, ktoré boli v porovnaní so všetkými bezplatnými kurzami dostupné len v kurze od spoločnosti Wolfram. Užívatelia môžu vyplniť testy ako v našej aplikácii, tak aj v kurze od Wolfram ľubovoľný počet krát, čo umožňuje overiť si svoje znalosti a v prípade negatívneho výsledku sa vrátiť a doštudovať si potrebné časti danej problematiky. Na rozdiel od testov, ktoré v našej aplikácii obsahujú iba 4 otázky, testy v kurze od spoločnosti Wolfram zahŕňajú až 10 otázok. Avšak, tieto otázky sú zostavené zo 4 až 5 tém. V našej aplikácii sú testy k dispozícii po každej kapitole a otázky sú vytvorené len z danej kapitoly. To znamená, že užívateľ si svoje znalosti overuje len z konkrétnej témy a nie z väčšieho okruhu tém. Tento prístup mu umožňuje získať hlbšie porozumenie o tom, ako dôkladne zvládol danú tému.

Obsahová napln našej aplikácie je primárne zameraná iba na témy, ktoré sú preberané na predmete Úvod do signálov a systémov. Vo väčšine dostupných kurzov je však k dispozícii širšie spektrum tém, ktoré umožňujú v porovnaní s našou aplikáciou hlbšie štúdium problematiky signálov a systémov. Nami vytvorená aplikácia je však škálovateľná a do budúcnosti je možné ju prípadne rozšíriť o ďalšie témy. Do budúcnosti je možné rozšíriť GUI o komplexnejšie témy, ako je napríklad číslicové spracovanie signálov a číslicové filtre. Taktiež by sme mohli do GUI doplniť viac testových otázok, ďalšie vizuálne a video ukážky a prípadne vytvoriť audio nahrávky, ktoré by si užívateľovi prehral pomocou tlačidla. Tieto nahrávky by mohli slúžiť na vysvetlenie daných ukážok alebo procesov demonštrujúcich problematiku signálov a systémov. V neposlednom rade by bolo výhodné, ak by GUI bolo prístupné prostredníctvom webového rozhrania, čo by umožnilo užívateľom prístup z akéhokoľvek zariadenia a bez potreby inštalácie. Tieto rozšírenia by prispeli k ďalšiemu zlepšeniu použiteľnosti, funkčnosti a všestrannosti GUI, čím by sa posilnil jeho prínos ako výučbového nástroja v oblasti signálov a systémov.

## ZÁVER

V tejto práci sme sa venovali vytvoreniu GUI pre popis metód spracovania signálov. Na vytvorenie GUI sme využili programovací jazyk MATLAB a rozšírili sme ho o webové jazyky, čo nám umožnilo vylepšiť jeho vizuálny aj funkčný aspekt. GUI sme navrhovali s úmyslom, aby slúžilo ako interaktívna výučbová pomôcka, a preto sme ho vybavili teóriou, vizuálnymi interaktívnymi ukážkami, testami, videami a príkladmi. Kombinácia týchto prvkov poskytuje užívateľom možnosť interaktívne sa ponoriť do problematiky signálov a systémov a lepšie si osvojiť dané koncepty. Výpočtové príklady sme doplnili o vizualizáciu pomocou jazyka MATLAB, čo nielenže uľahčuje pochopenie daného príkladu, ale taktiež umožňuje používateľom získavať skúsenosti a zdokonaľovať sa v programovaní v tomto jazyku. Pred vytvorením samotných príkladov sme museli do GUI implementovať kódový editor, ktorý sme vytvorili za pomoci webových jazykov. Tento editor umožňuje používateľom pohodlne písať a upravovať kód priamo v GUI prostredí, čo zlepšuje užívateľské pohodlie a zvyšuje efektivitu učenia. Vizuálne ukážky a videá boli vytvorené s cieľom preskúmať a demonštrovať konkrétne princípy, metódy alebo procesy. Tieto prostriedky vizualizácie umožňujú užívateľom interaktívne sa oboznámiť s danou problematikou a lepšie si ju predstaviť a porozumieť.

Výsledkom tejto práce je interaktívne a responzívne GUI, ktoré sa zameriava primárne na predmet Úvod do signálov a systémov. Avšak, obsahuje materiály, ktoré sú využiteľné nielen v tomto konkrétnom predmete, ale aj v akomkoľvek inom, ktorý sa zaoberá problematikou signálov a systémov. Vytvorené GUI poskytuje užívateľom prístup k bohatej škále obsahu, čo ho robí významným študijným zdrojom pre študentov a záujemcov v tejto oblasti.

Pre zvýšenie dostupnosti sme GUI preložili aj do anglického jazyka, aby ho mohli využívať aj zahraniční študenti. Ďalej sme zabezpečili jednoduchší prístup a možnosť zdieľania tým, že sme vytvorené GUI umiestnili na cloudové platformy MATLAB File Exchange a GitHub, odkiaľ si ho môžu užívatelia stiahnuť a následne nainštalovať.

Vytvorené GUI má určite svoje nedokonalosti a obsahuje chyby, ktoré sa nám žiaľ nepodarilo odstrániť. Tieto chyby sú spôsobené rozsiahlosťou aplikácie a zvoleným vývojovým prostredím. Vzniknuté chyby ale nie sú fatálne a neovplyvňujú funkcionality GUI, len si vyžadujú určitú dávku trpezlivosti zo strany užívateľa.

# ZOZNAM POUŽITEJ LITERATÚRY

- [1] FRÁNEKOVÁ, M – MUZIKÁŘOVÁ, L. Teória informácie a signálov. Žilina: EDIS, 2009. 217 s. ISBN 978-80-554-0075-4.
- [2] ONDRÁČEK, O. Signály a sústavy. Bratislava: STU, 2008. 341 s. ISBN 978-80-227-2956-7.
- [3] GLADIŠOVÁ, I. – MIHALÍK, J. Spojité signály. Košice: LČSOV FEI TU, 2015. 65 s. ISBN 978-80-533-1936-0.
- [4] CHMÚRNY, J. – ONDRÁČEK, O. – ŽIDEK, F. Signály a sústavy pre odbor rádioelektronika. Bratislava: Edičné stredisko SVŠT, 1991. 303 s. ISBN 80-227-0356-7.
- [5] TICHÁ, D. – VOLNER, R. Signály a šumy. Žilina: EDIS, 2003. 315 s. ISBN 80-8070-147-4.
- [6] PRCHAL, J. Signály a soustavy. Praha: SNTL, 1987. 324 s. ISBN 660-69-556-4506-4.
- [7] MATHWORKS, What Is convolution?. [Online]. [cit. 2023-10-05]. Dostupné na: <https://www.mathworks.com/discovery/convolution.html>.
- [8] SMÉKAL, Z. – ŠEBESTA, V. Signály a soustavy. Přednášky. Brno: UREL FEKT VUT v Brně, 2003. 165 s. ISBN 80-214-2434- 6.
- [9] HAMID NAWAB, S. – OPPENHEIM, A. – WILLISKY, A. Signals & Systems Second Edition. USA, New Jersey: PRENTICE HALL, 1996. 957 s. ISBN 0-13-814757-4.
- [10] MITOPENCOURSEWARE, Signals And Systems. [Online]. [cit. 2023-11-05]. Dostupné na: <https://ocw.mit.edu/courses/res-6-007-signals-and-systems-spring-2011/pages/introduction/>.
- [11] SUPERLECTURES, ISS Signály a systémy. [Online]. [cit. 2023-11-05]. Dostupné na: <https://www.superlectures.com/fit-iss/>.
- [12] WOLFRAM U, All Classes and Courses. [Online]. [cit. 2023-11-07]. Dostupné na: <https://www.wolfram.com/wolfram-u/courses/catalog/page/2/>.
- [13] WOLFRAM U, Signals, Systems and Signal Processing. [Online]. [cit. 2023-11-07]. Dostupné na: <https://www.wolfram.com/wolfram-u/courses/image-signal-processing/signals-systems-and-signal-processing/>.
- [14] WOLFRAM, Wolfram Language. [Online]. [cit. 2023-11-07]. Dostupné na: <https://www.wolfram.com/language/>.
- [15] WOLFRAM, Signal Processing. [Online]. [cit. 2023-11-07]. Dostupné na: <https://reference.wolfram.com/language/guide/SignalProcessing.html>.
-



- [16] WOLFRAM U, An Elementary Introduction to the Wolfram Language. [Online]. [cit. 2023-11-07]. Dostupné na : <https://www.wolfram.com/wolfram-u/courses/wolfram-language/an-elementary-introduction-to-the-wolfram-language/>.
- [17] PYTHON DOCUMENTATION, General Python FAQ. [Online]. [cit. 2023-11-07]. Dostupné na: <https://docs.python.org/3/faq/general.html#what-is-python>.
- [18] LUTZ, M. Learning Python Fourth Edition. USA, Kalifornia: O'Reilly Media, 2009. 1600 s. ISBN: 978-0-596-15806-4.
- [19] SCIPY, SciPy documentation. [Online]. [cit. 2023-11-07]. Dostupné na: <https://docs.scipy.org/doc/scipy/>.
- [20] NUMPY, NumPy documentation. [Online]. [cit. 2023-11-07]. Dostupné na: <https://numpy.org/doc/stable/>.
- [21] MATPLOTLIB, Matplotlib: Visualization with Python. [Online]. [cit. 2023-11-07]. Dostupné na: <https://matplotlib.org/>.
- [22] MCFARLAND, A. 10 Best Python Libraries for GUI. [Online]. [cit. 2023-11-07]. Dostupné na: <https://www.unite.ai/10-best-python-libraries-for-gui/>.
- [23] KOLÁČEK, J. – KONEČNÁ, K. Jak pracovat s MATLABem. [Online]. [cit. 2023-11-07]. Dostupné na: <https://www.math.muni.cz/~kolacek/vyuka/vypsyst/navod.pdf>.
- [24] MATHWORKS, What Is MATLAB?. [Online]. [cit. 2023-12-05]. Dostupné na: <https://www.mathworks.com/discovery/what-is-matlab.html>.
- [25] MATHWORKS, MATLAB – Functions. [Online]. [cit. 2023-12-05]. Dostupné na: [https://www.mathworks.com/help/matlab/referencelist.html?type=function&category=index&s\\_tid=CRUX\\_lftnav\\_function\\_index](https://www.mathworks.com/help/matlab/referencelist.html?type=function&category=index&s_tid=CRUX_lftnav_function_index).
- [26] MATHWORKS, What Is MATLAB GUI?. [Online]. [cit. 2023-12-05]. Dostupné na: <https://www.mathworks.com/discovery/matlab-gui.html>.
- [27] GNU OCTAVE, About. [Online]. [cit. 2023-12-05]. Dostupné na: <https://octave.org/about>.
- [28] PEDAMKAR, P. Matlab vs Octave. [Online]. [cit. 2023-12-05]. Dostupné na: <https://www.educba.com/matlab-vs-octave/>.
- [29] OCTAVE FORGE, signal. [Online]. [cit. 2023-12-05]. Dostupné na: <https://octave.sourceforge.io/signal/index.html>.
- [30] OCTAVE DOCUMENTATION, Signal Processing. [Online]. [cit. 2023-12-05]. Dostupné na: <https://docs.octave.org/v4.0.1/Signal-Processing.html>.
- [31] OCTAVE DOCUMENTATION, GUI Development. [Online]. [cit. 2023-12-05]. Dostupné na: <https://docs.octave.org/v6.3.0/GUI-Development.html>.
-

# ČESTNÉ VYHLÁSENIE

Vyhlasujem, že som zadanú bakalársku prácu vypracoval samostatne, pod odborným vedením vedúceho bakalárskej práce doc. Ing. Patrik Kamencay, PhD. a používal som len literatúru uvedenú v práci.

V Žiline dňa 13. mája 2024

Adam Petrovič

---

## PRÍLOHOVÁ ČASŤ

---

# ZOZNAM PRÍLOH

Príloha A ..... II

## Príloha A

Príloha na USB obsahuje:

- Zdrojové kódy a súbory slovenskej a anglickej verzie aplikácie.
- Vyexportovanú aplikáciu v slovenskej a anglickej verzii vo formáte *.exe*.
- Vyexportovanú aplikáciu v slovenskej a anglickej verzii vo formáte MATLAB aplikácie (*.mlappinstall*).
- Elektronickú formu bakalárskej práce.